

UNIVERSIDAD POLITÉCNICA DE MADRID
FACULTAD DE INFORMÁTICA
Departamento de Lenguajes, Sistemas Informáticos e
Ingeniería del Software



Tesis Doctoral

**Técnicas para la Estimación y Planificación de Proyectos de Software
con Ciclos de Vida Incremental y Paradigma Orientado a Objetos**

Autor: **José Antonio Pow Sang Portillo**

Directores: Ana María Moreno Sánchez-Capuchino
Ricardo Imbert Paredes

Madrid, marzo de 2012

A mis padres...

Agradecimientos

El desarrollar una tesis doctoral es un esfuerzo personal muy grande; y, para poder concluirla, se necesita el apoyo de muchas personas. Por ello, quiero expresar en estas líneas mi sincero agradecimiento a todas aquellas personas que Dios puso en mi camino para que pudiera concluir este trabajo.

En primer lugar, quiero agradecer a mis directores de tesis, Ana y Ricardo, quienes, a pesar de lo complicado de dirigir una tesis estando tan lejos de España, me mostraron siempre su disposición y dedicaron su valioso tiempo para orientarme.

A Natalia, quien me apoyó para poder iniciar a escribir el documento de tesis y enlazar todos los trabajos previos que había realizado, orientación que fue muy valiosa para mí. Aunque no has visto el trabajo final, te considero como mi tercera directora de tesis.

A las autoridades de mi universidad, quienes me apoyaron para poder realizar los viajes a España y poder terminar la tesis. En especial a Domingo, muchas gracias.

A Arturo, mi gran amigo desde que estudiamos Ingeniería Informática en la PUCP, por su apoyo para escribir los artículos en inglés relacionados a este trabajo. A Nelson, muchas gracias por tus consejos para la defensa, los tendré muy en cuenta. A Loretta, por su gran apoyo en la parte estadística de la tesis y por darme ánimos para terminar. A Raúl, por todas las gestiones que tuviste que hacer por mi examen de suficiencia investigadora, apoyo que no sé cómo agradecerle. A Jackeline, muchas gracias por los trámites que necesitaba realizar para finalizar el doctorado.

A Lourdes, Luisa, Emilio, José Angel, Santiago y Yoli, a quienes considero como mi familia. Muchas gracias por todo su apoyo cada vez que venía a Madrid y por todo su cariño no solo para mí sino también para Milagros y Gabriel.

Finalmente, a mi esposa, Milagros, por su cariño, paciencia y comprensión. A mi hijo, Gabriel, quien a pesar de su corta edad, tres años, supo comprender que no podía jugar con él porque estaba haciendo mi tarea y que si no terminaba "me iba a ir con el padre del perro del respeto". A los dos, por las horas que les dejé de dedicar. A mi tía Chela y mi tío Miguel, quienes con mucho cariño, cuidaban a Gabrielito para que pudiera avanzar con la tesis. A mi hermana Mary, por su apoyo en mis estudios, desde la primaria.

Resumen

La estimación de costos y esfuerzos sigue siendo una de las tareas más difíciles en la gestión de un proyecto de software. Esta actividad es realizada por el jefe de proyecto, quien es responsable de hacer dichas estimaciones lo más precisas posible. En la actualidad existen técnicas que permiten realizar esta labor aunque, lamentablemente, aún no hay técnicas maduras específicas para enfoques de desarrollo como la orientación a objetos o los sistemas expertos. A ello se suma el problema de la escasa información proporcionada por las técnicas de estimación existentes para su aplicación a ciclos de vida de desarrollo de software diferentes al de cascada, como, por ejemplo, los ciclos de vida incremental o iterativo.

Debido a la problemática existente, se diseñó Tupuy que es un conjunto de técnicas que apoya en la estimación y planificación basada en Puntos de función para proyectos de desarrollo de software orientados a objetos que empleen un modelo de ciclo de vida incremental. Esta propuesta está conformada por tres técnicas: UML₂FP, *Use Case Precedence Diagram (UCPD)* e Incremental-FP.

UML₂FP es una técnica que permite realizar el cálculo de Puntos de función con modelos orientados a objetos y UCPD apoya en la priorización o definición de la secuencia de construcción considerando la perspectiva o punto de vista del desarrollador en términos de facilidad de construcción. Con los resultados que se obtienen con UML₂FP y UCPD, la técnica Incremental-FP, permite definir qué casos de uso se van a construir en cada incremento y estimar el esfuerzo que se requiere para desarrollar cada incremento.

La evaluación de UML₂FP y UCPD se realizó mediante experimentos controlados con alumnos de pregrado, alumnos de posgrado y profesionales con experiencia en la industria, con resultados alentadores. En el caso de Incremental-FP, la evaluación se realizó mediante proyectos de desarrollo de software con alumnos del cuarto año del programa de pregrado en Ingeniería Informática de la Pontificia Universidad Católica del Perú; habiéndose obtenido, generalmente, una diferencia menor al 20% entre el esfuerzo estimado y el esfuerzo real. Adicionalmente, en los experimentos controlados para evaluar UCPD, se pudo determinar que las relaciones propuestas por el *Method Adoption Model* de Moody sí se pueden confirmar con las muestras obtenidas con profesionales, pero no con alumnos de pregrado; a pesar de que los resultados cuantitativos fueron similares. Esto nos podría sugerir que la evaluación cualitativa de técnicas con alumnos de pregrado deben tomarse con precaución, mas no la cuantitativa.

Abstract

The estimation of effort and cost is still one of the hardest tasks in software project management. This activity is performed by the project manager who is responsible for making such estimations as accurate as possible. Currently, there are techniques to do this work, but unfortunately there are no specific mature approaches to develop object-oriented software or expert systems. Also, there is limited information provided by the techniques to utilize them in non-waterfall life cycle models such as incremental or iterative.

To address this situation, it was designed Tupuy which is a set of techniques that supports Function Point-based estimation and planning for software development projects that utilize object-oriented and incremental life cycle models. Tupuy consists of three techniques named UML₂FP, Use Case Precedence Diagram (UCPD) and Incremental-FP.

UML₂FP is a technique that allows Function Points counting based on object-oriented models, and UCPD is a technique that addresses the problem of determining the construction sequence or prioritization of a software product from the developer's perspective in terms of ease of construction. With the results obtained through the application of UML₂FP and UCPD, Incremental-FP defines which use case has to be constructed in which increment, and estimates the effort required to develop each increment.

UML₂FP and UCPD evaluation was performed using controlled experiments with undergraduate students, graduate students and practitioners with experience in the industry. The results obtained were satisfactory. The Incremental-FP evaluation was performed with undergraduate students from software development projects of a fourth-year course in Informatics Engineering at Pontificia Universidad Católica del Perú. In this case, the difference between the estimated effort and actual effort was usually less than 20%. In addition, from the data obtained with the controlled experiments to evaluate UCPD, it was determined that the relationships proposed by the Moody's Method Adoption Model could be confirmed with practitioners, but not with undergraduate students. These results may suggest that the qualitative assessments of techniques with undergraduate students should be taken with caution, but not the quantitative assessments.

Índice General

1. Introducción.....	1
1.1 Problema de Investigación.....	1
1.2 Área de Investigación.....	9
1.3 Objetivos y Aproximación de la Solución	13
1.4 Estructura del Documento	15
2. Estado de la Cuestión	17
2.1 Revisión Sistemática en Ingeniería de Software	17
2.2 Definición de Preguntas de Investigación	19
2.3 Estrategia de Búsqueda y Selección.....	20
2.4 Proceso de búsqueda y selección.....	25
2.5 Redefinición de la Estrategia de Búsqueda y Nuevo Proceso de Búsqueda.....	26
2.6 Técnicas para Calcular Puntos de Función en Modelos Orientados a Objetos	38
2.7 Técnicas para Calcular el Esfuerzo Requerido por cada Incremento	58
2.8 Técnicas para priorizar o definir la secuencia de construcción de casos de uso	62
2.9 Amenazas a la Validez de la Revisión Sistemática.....	70
2.10 Conclusiones del Estado de la Cuestión	72
3. Planteamiento del Problema	77
3.1 Definición del Problema.....	77
3.2 Hipótesis de Trabajo	79
3.3 Descripción de la Solución Propuesta.....	81
3.4 Estructura de la Resolución.....	82
4. Aproximación a la Solución: Tupuy	83
4.1 Conceptos en los que se Fundamenta la Aproximación a la Solución	83
4.2 Descripción de TUPUY	99
4.3 Ejemplo de la aplicación de Tupuy	119
4.4 Evaluación de TUPUY	127
5. Evaluación de la Conversión a Ficheros de UML₂FP	129
5.1 Visión General de la Evaluación.....	129
5.2 Materiales Empleados en los Experimentos	131
5.3 Tareas Realizadas Durante los Experimentos.....	132
5.4 Resultados	133
5.5 Amenazas a la Validez de los Experimentos	150
5.6 Conclusiones de la Evaluación de las Reglas de Conversión a Ficheros de UML ₂ FP	151
6. Evaluación del Diagrama de Precedencias de Casos de Uso	155
6.1 Visión General de la Evaluación.....	155
6.2 EI-PREG: Evaluación cualitativa con alumnos de pregrado	157
6.3 Evaluación mediante experimentos controlados	165
6.4 Evaluación de la percepción de UCPD de los experimentos controlados	196
6.5 Evaluación del MAM	203
6.6 Conclusiones de la Evaluación de UCPD	228
7. Evaluación de Incremental-FP.....	231

7.1	Visión General de la Evaluación	231
7.2	Información de los proyectos.....	233
7.3	Resultados Obtenidos	236
7.4	Conclusiones de la Evaluación de Incremental-FP	244
8.	Conclusiones y Líneas de Trabajo a Futuro.....	247
8.1	Tupuy	247
8.2	UML ₂ FP.....	248
8.3	Use Case Precedence Diagram	250
8.4	Incremental-FP	251
8.5	Publicación de Resultados	253
	Bibliografía	255
	Anexo A.....	267
A.1	Cadenas de Búsqueda para la Pregunta 2.....	267
A.2	Cadenas de Búsqueda para la Pregunta 2a	269
A.3	Cadenas de Búsqueda para la Pregunta 2b.....	270
A.4	Cadenas de Búsqueda para la Pregunta 2c.....	271
A.5	Resultados obtenidos para la búsqueda de la pregunta 2c.....	272
	Anexo B.....	277
B.1	Diagramas de Casos de Uso.....	277
B.2	Actualizar Producto	277
B.3	Actualizar Cliente.....	278
B.4	Registrar Venta	279
B.5	Anular Venta	280
B.6	Ver Reporte de Ventas.....	280
	Anexo C.....	283
C.1	Evaluación con Profesionales: EC1-PROF.....	283
C.2	Evaluación con Docentes: EP2-DOC	298
C.3	Evaluación con Estudiantes de Pregrado: EC3-PREG.....	305
C.4	Evaluación con Estudiantes de Posgrado: EC4-POST.....	310
C.5	Evaluación con Estudiantes de Posgrado: EC5-POST.....	312
	Anexo D	317
D.1	Evaluación con Alumnos de Pregrado: EI-PREG.....	317
D.2	Evaluación con Profesionales: EC1-PROF.....	320
D.3	Evaluación con Estudiantes de Posgrado: EC2-POST.....	343
D.4	Evaluación con Estudiantes de Pregrado: EC3-PREG.....	348
	Anexo E.....	353
E.1	Información del Proyecto A del 2004-2.....	353
E.2	Información del Proyecto B del 2004-2.....	355

Índice Detallado

1. Introducción.....	1
1.1 Problema de Investigación.....	1
1.1.1 El contexto del problema de investigación	1
1.1.1.1 Los sistemas de información.....	1
1.1.1.2 Puntos de función	3
1.1.1.3 Los modelos de ciclo de vida.....	4
1.1.2 Relevancia del problema de investigación y dificultades que plantea	8
1.2 Área de Investigación	9
1.3 Objetivos y Aproximación de la Solución	13
1.4 Estructura del Documento	15
2. Estado de la Cuestión	17
2.1 Revisión Sistemática en Ingeniería de Software	17
2.2 Definición de Preguntas de Investigación	19
2.3 Estrategia de Búsqueda y Selección.....	20
2.3.1 Términos de búsqueda.....	20
2.3.2 Fases del proceso de búsqueda	23
2.3.3 Criterios de inclusión y exclusión.....	24
2.3.4 Selección de artículos.....	25
2.4 Proceso de búsqueda y selección	25
2.5 Redefinición de la Estrategia de Búsqueda y Nuevo Proceso de Búsqueda.....	26
2.5.1 Pregunta de investigación 2(a): Técnicas basadas en Puntos de función para modelos orientado a objetos.....	27
2.5.2 Pregunta de investigación 2(b): Técnicas para calcular el esfuerzo requerido por cada incremento.....	31
2.5.3 Pregunta de investigación 2(c): Técnicas para priorizar o definir la secuencia de la construcción de casos de uso	34
2.6 Técnicas para Calcular Puntos de Función en Modelos Orientados a Objetos	38
2.6.1 Ficheros de Puntos de Función	40
2.6.1.1 Clases y atributos.....	41
2.6.1.2 Relaciones de asociación, agregación y composición entre clases	42
2.6.1.3 Relaciones de herencia entre clases	45
2.6.1.4 Clase asociación.....	47
2.6.1.5 Resumen de la revisión de ficheros	49
2.6.2 Transacciones de Puntos de función	50
2.6.2.1 Diagramas de UML empleados y metodologías de desarrollo de software.....	51
2.6.2.2 Identificación de tipo de transacciones	54
2.6.2.3 Reglas adicionales para identificar DET	55
2.6.2.4 Resumen de la revisión de transacciones.....	56
2.6.3 Conclusiones de la revisión de técnicas para calcular puntos de función en modelos orientados a objetos.....	57
2.7 Técnicas para Calcular el Esfuerzo Requerido por cada Incremento	58
2.7.1 Priorización de casos de uso y métricas para estimar el tamaño de software	59
2.7.2 Modelo de conversión de Puntos de función a orientación a objetos	59
2.7.3 Estimación del esfuerzo entre incrementos o iteraciones	59
2.7.4 Conclusiones de la revisión de técnicas para calcular el esfuerzo requerido por cada incremento.....	60
2.8 Técnicas para priorizar o definir la secuencia de construcción de casos de uso	62

2.8.1	Utilización de casos de uso o técnica equivalente	66
2.8.2	Precondiciones.....	67
2.8.3	Secuencias entre requisitos	67
2.8.4	Notación gráfica y notación UML.....	68
2.8.5	Conclusiones de la revisión de técnicas para priorizar la secuencia de construcción de casos de uso	69
2.9	Amenazas a la Validez de la Revisión Sistemática	70
2.9.1	Bases de datos empleadas y "literatura gris".....	70
2.9.2	Parcialidad en la selección de artículos	71
2.9.3	Exclusión de artículos relevantes.....	71
2.9.4	Términos empleados en las búsquedas.....	72
2.10	Conclusiones del Estado de la Cuestión.....	72
2.10.1	Búsqueda y selección de artículos	72
2.10.2	Cálculo de Puntos de función en modelos orientados a objetos	73
2.10.3	Cálculo del esfuerzo requerido por cada incremento	74
2.10.4	Definición de la priorización o secuencia de construcción de casos de uso	75
3.	Planteamiento del Problema.....	77
3.1	Definición del Problema	77
3.2	Hipótesis de Trabajo	79
3.3	Descripción de la Solución Propuesta	81
3.4	Estructura de la Resolución	82
4.	Aproximación a la Solución: Tupuy.....	83
4.1	Conceptos en los que se Fundamenta la Aproximación a la Solución	83
4.1.1	La especificación de requisitos con casos de uso	83
4.1.1.1	Actores	84
4.1.1.2	Los escenarios y los casos de uso	84
4.1.1.3	La documentación de los casos de uso	85
4.1.1.4	Las relaciones entre caso de uso	86
4.1.1.5	Casos de uso concretos y casos de uso abstractos	88
4.1.2	El análisis con diagramas de clase	89
4.1.3	Estimación del esfuerzo con Cocomo II y Puntos de Función.....	94
4.2	Descripción de TUPUY.....	99
4.2.1	UML ₂ PF.....	100
4.2.1.1	Cálculo de ficheros.....	100
4.2.1.2	Cálculo de transacciones	106
4.2.2	Use Case Precedence Diagram	111
4.2.3	Incremental-FP.....	114
4.2.3.1	Estimar el tamaño de cada caso de uso (actividad 3).....	115
4.2.3.2	Definir incrementos y esfuerzo requerido (actividad 4)	117
4.2.3.3	Evaluar resultados de la construcción del incremento (actividad 6)	118
4.3	Ejemplo de la aplicación de Tupuy	119
4.3.1	Aplicación de UML ₂ FP	120
4.3.2	Aplicación de UCPD	122
4.3.3	Aplicación de Incremental-FP	123
4.3.3.1	Estimación del tamaño de cada caso de uso (actividad 3).....	123
4.3.3.2	Definición de incrementos y esfuerzo requerido (actividad 4)	125
4.3.3.3	Evaluación de resultados de la construcción del incremento (actividad 6) ..	126
4.4	Evaluación de TUPUY.....	127
5.	Evaluación de la Conversión a Ficheros de UML₂FP.....	129
5.1	Visión General de la Evaluación	129
5.2	Materiales Empleados en los Experimentos.....	131

5.3	Tareas Realizadas Durante los Experimentos	132
5.4	Resultados	133
5.4.1	<i>EC1-PROF: Primer experimento controlado con profesionales.....</i>	<i>135</i>
5.4.2	<i>EP2-DIC: Experimento piloto con docentes y asistentes de docencia</i>	<i>137</i>
5.4.3	<i>EC3-PREG: Tercer experimento controlado con estudiantes de pregrado ...</i>	<i>139</i>
5.4.4	<i>EC4-POST: Cuarto experimento controlado con estudiantes de posgrado ...</i>	<i>143</i>
5.4.5	<i>EC5-POST: Quinto experimento controlado con estudiantes de posgrado ...</i>	<i>146</i>
5.4.6	<i>Resultados del cuestionario sobre percepción de las técnicas.....</i>	<i>148</i>
5.5	Amenazas a la Validez de los Experimentos	150
5.5.1	Validez de la construcción.....	150
5.5.2	Validez interna.....	150
5.5.3	Validez externa	151
5.6	Conclusiones de la Evaluación de las Reglas de Conversión a Ficheros de UML ₂ FP	151
6.	Evaluación del Diagrama de Precedencias de Casos de Uso	155
6.1	Visión General de la Evaluación.....	155
6.2	EI-PREG: Evaluación cualitativa con alumnos de pregrado	157
6.2.1	<i>Sujetos</i>	<i>158</i>
6.2.2	<i>Materiales.....</i>	<i>159</i>
6.2.3	<i>Resultados.....</i>	<i>160</i>
6.2.3.1	Facilidad de uso percibida.....	162
6.2.3.2	Utilidad percibida	162
6.2.3.3	Intención de uso	164
6.2.4	<i>Discusión de los resultados</i>	<i>164</i>
6.3	Evaluación mediante experimentos controlados	165
6.3.1	<i>EC1-PROF: Primer experimento controlado realizado con profesionales.....</i>	<i>167</i>
6.3.1.1	Materiales y casos de estudio	167
6.3.1.2	Tareas realizadas durante el experimento.....	168
6.3.1.3	Resultados cuantitativos.....	169
6.3.2	<i>EC2-POST: Segundo experimento controlado con estudiantes de posgrado</i>	<i>183</i>
6.3.2.1	Materiales y casos de estudio	184
6.3.2.2	Tareas realizadas durante el experimento.....	184
6.3.2.3	Resultados cuantitativos.....	186
6.3.3	<i>EC3-PREG: Tercer experimento controlado con estudiantes de pregrado ...</i>	<i>189</i>
6.3.4	<i>Amenazas a la validez.....</i>	<i>193</i>
6.3.4.1	Validez de la construcción.....	193
6.3.4.2	Validez interna.....	193
6.3.4.3	Validez externa	194
6.3.5	<i>Discusión de los resultados</i>	<i>195</i>
6.4	Evaluación de la percepción de UCPD de los experimentos controlados	196
6.4.1	<i>Evaluación con profesionales.....</i>	<i>196</i>
6.4.2	<i>Evaluación con estudiantes de pregrado</i>	<i>200</i>
6.4.3	<i>Discusión de los resultados</i>	<i>203</i>
6.5	Evaluación del MAM	203
6.5.1	<i>Evaluación con estudiantes de pregrado</i>	<i>205</i>
6.5.1.1	Modelo de regresión para la relación utilidad percibida-intención de uso para EI-PREG	207
6.5.1.2	Modelo de regresión para la relación utilidad percibida-intención de uso para EC3-PREG	210
6.5.2	<i>Evaluación con profesionales.....</i>	<i>213</i>
6.5.2.1	Modelo de regresión para la relación facilidad de uso-utilidad para EC1-PROF.	214
6.5.2.2	Modelo de regresión para la relación facilidad de uso-utilidad para EC2-POST.	218

6.5.2.3	Modelo de regresión para la relación facilidad de uso-utilidad vs. intención de uso para EC1-PROF.....	221
6.5.2.4	Modelo de regresión para la relación facilidad de uso-utilidad vs. intención de uso para EC2-POST.....	224
6.5.3	<i>Discusión de los resultados</i>	228
6.6	Conclusiones de la Evaluación de UCPD.....	228
7.	Evaluación de Incremental-FP	231
7.1	Visión General de la Evaluación	231
7.2	Información de los proyectos.....	233
7.2.1	<i>Conocimiento de los estudiantes y la distribución de los equipos</i>	233
7.2.2	<i>Características de los proyectos</i>	234
7.2.3	<i>Recolección de datos</i>	235
7.3	Resultados Obtenidos.....	236
7.3.1	<i>Segundo incremento</i>	238
7.3.2	<i>Tercer incremento</i>	239
7.3.3	<i>Comparación de Incremental-FP con estimaciones globales</i>	241
7.3.4	<i>Comparación de Incremental-FP y estimación con puntos de función ajustados</i>	243
7.4	Conclusiones de la Evaluación de Incremental-FP	244
8.	Conclusiones y Líneas de Trabajo a Futuro.....	247
8.1	Tupuy	247
8.2	UML ₂ FP.....	248
8.3	Use Case Precedence Diagram	250
8.4	Incremental-FP	251
8.5	Publicación de Resultados	253
	Bibliografía	255
	Anexo A.....	267
A.1	Cadenas de Búsqueda para la Pregunta 2.....	267
A.2	Cadenas de Búsqueda para la Pregunta 2a	269
A.3	Cadenas de Búsqueda para la Pregunta 2b.....	270
A.4	Cadenas de Búsqueda para la Pregunta 2c.....	271
A.5	Resultados obtenidos para la búsqueda de la pregunta 2c.....	272
	Anexo B.....	277
B.1	Diagramas de Casos de Uso	277
B.2	Actualizar Producto	277
B.3	Actualizar Cliente.....	278
B.4	Registrar Venta	279
B.5	Anular Venta	280
B.6	Ver Reporte de Ventas.....	280
	Anexo C.....	283
C.1	Evaluación con Profesionales: EC1-PROF.....	283
C.1.1	<i>Casos</i>	283
C.1.2	<i>Formularios y cuestionario</i>	290
C.1.3	<i>Tablas con información obtenida de los formularios y del cuestionario</i>	293
C.1.4	<i>Cuestionarios llenados y diagramas realizados por un participante del estudio</i>	295
C.2	Evaluación con Docentes: EP2-DOC	298
C.2.1	<i>Casos</i>	298

C.2.2	Tablas con información obtenida de los formularios y del cuestionario	305
C.3	Evaluación con Estudiantes de Pregrado: EC3-PREG	305
C.4	Evaluación con Estudiantes de Posgrado: EC4-POST	310
C.5	Evaluación con Estudiantes de Posgrado: EC5-POST	312
Anexo D	317
D.1	Evaluación con Alumnos de Pregrado: EI-PREG	317
D.2	Evaluación con Profesionales: EC1-PROF	320
D.2.1	Casos de estudio.....	321
D.2.2	Cuestionarios.....	328
D.2.3	Tablas con información obtenida de los cuestionarios.....	332
D.2.4	Cuestionarios llenados y diagramas realizados por los participantes	335
D.3	Evaluación con Estudiantes de Posgrado: EC2-POST	343
D.3.1	Cuestionarios.....	343
D.3.2	Tablas con información obtenida de los cuestionarios.....	346
D.4	Evaluación con Estudiantes de Pregrado: EC3-PREG	348
Anexo E	353
E.1	Información del Proyecto A del 2004-2	353
E.2	Información del Proyecto B del 2004-2	355

Índice de Tablas

Tabla 1-1. Tareas del Swebok consideradas en la investigación	10
Tabla 1-2. Áreas del conocimiento del Pmbok consideradas en la investigación	12
Tabla 2-1. Diferencias entre revisiones tradicionales y sistemáticas.....	18
Tabla 2-2. Resultados de la aplicación del criterio PICOC	19
Tabla 2-3. Términos en inglés derivados del criterio PICOC.....	20
Tabla 2-4. Términos relacionados para <i>population e intervention</i>	21
Tabla 2-5. Resultados de búsqueda inicial	25
Tabla 2-6. Resultados de la aplicación del criterio PICOC para la pregunta 2(a).....	27
Tabla 2-7. Cantidad de artículos encontrados y seleccionados para la pregunta 2(a)	29
Tabla 2-8. Artículos seleccionados para la pregunta 2(a)	29
Tabla 2-9. Resultados de la aplicación del criterio PICOC para la pregunta 2(b)	31
Tabla 2-10. Cantidad de artículos encontrados y seleccionados para la pregunta 2(b) ..	33
Tabla 2-11. Artículos seleccionados para la pregunta 2(b)	33
Tabla 2-12. Términos en inglés para <i>population e intervention</i> para la pregunta 2(c)	35
Tabla 2-13. Términos en inglés derivados para <i>population e intervention</i> para la pregunta 2(c)	35
Tabla 2-14. Cantidad de artículos encontrados y seleccionados para la pregunta 2(c)...	37
Tabla 2-15. Técnicas identificadas para Puntos de función y modelos orientados a objetos	38
Tabla 2-16. Reglas para determinar ficheros en PF (tomado de [Ifpug, 2009])	40
Tabla 2-17. Propuestas de las técnicas para asociación	43
Tabla 2-18. Propuestas de las técnicas para agregación.....	43
Tabla 2-19. Propuestas de las técnicas para composición	45
Tabla 2-20. Propuestas de las técnicas para herencia.....	47
Tabla 2-21. Tabla comparativa con el grado de cumplimiento de las técnicas de PF para determinar los ficheros	50
Tabla 2-22. Reglas para determinar DET en las transacciones de PF	51
Tabla 2-23. Propuestas de la técnicas para identificar y contar transacciones	52
Tabla 2-24. Diagramas empleados y metodología relacionada	53
Tabla 2-25. Propuestas para identificar tipos de transacciones.....	55
Tabla 2-26. Propuestas para identificar tipos de transacciones y reglas 2 y 3 para identificar DET	56
Tabla 2-27. Tabla comparativa de lo que proponen las técnicas para determinar las transacciones de PF	57

Tabla 2-28. Técnicas identificadas para cálculo de esfuerzo requerido para cada incremento	58
Tabla 2-29. Tabla comparativa entre técnicas para estimación del esfuerzo para cada iteración o incremento.....	61
Tabla 2-30. Publicaciones que consideran las dependencias o precedencias de casos de uso o requisitos	63
Tabla 2-31. Técnicas para determinar dependencias o precedencias de casos de uso o requisitos	65
Tabla 2-32. Técnicas que emplean casos de uso o equivalente	66
Tabla 2-33. Técnicas que emplean notación gráfica y UML	69
Tabla 2-34. Tabla comparativa entre técnicas para definir secuencia de construcción de casos de uso	70
Tabla 4-1. <i>Effort Multipliers</i> para el modelo de Post-Arquitectura de Cocomo II.....	95
Tabla 4-2. Reglas para asociación y agregación de UML ₂ FP	102
Tabla 4-3. Tabla para determinar la complejidad de ILF o EIF.....	104
Tabla 4-4. Tabla para determinar el peso en PFSA de ILF o EIF	104
Tabla 4-5. PFSA por transacción de cada caso de uso del ejemplo	122
Tabla 4-6. PFSA debido a ficheros por cada transacción del ejemplo	123
Tabla 4-7. PFSA de cada caso de uso del ejemplo	124
Tabla 4-8. Casos de uso que se implementarán por cada incremento en el ejemplo	125
Tabla 4-9. Estimación para el primer incremento del ejemplo	125
Tabla 4-10. Estimación para el segundo incremento del ejemplo	126
Tabla 4-11. Estimación para el tercer incremento del ejemplo	126
Tabla 5-1. Relación de experimentos controlados empleados para evaluar la conversión a ficheros de UML ₂ FP.....	130
Tabla 5-2. Tareas realizadas por los participantes de los experimentos controlados para la conversión a ficheros UML ₂ FP.....	133
Tabla 5-3. Ejemplo de resultados obtenidos por categoría en experimentos controlados para evaluar UML ₂ FP.....	134
Tabla 5-4. Resultados obtenidos por cada participante en EC1-PROF.....	135
Tabla 5-5. Valores estadísticos de los resultados obtenidos en EC1-PROF.....	135
Tabla 5-6. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC1-PROF.....	136
Tabla 5-7. Resultados de la prueba de los rangos señalados de Wilcoxon en EC1-PROF por categoría	136
Tabla 5-8. Resultados obtenidos por cada participante en EC2-DOC	138
Tabla 5-9. Valores estadísticos de los resultados obtenidos en EC2-DOC	138
Tabla 5-10. Situación 1 que se empleó originalmente en EC1-PROF y su actualización.	139

Tabla 5-11. Conocimientos y experiencia de los estudiantes de pregrado que participaron en EC3-PREG.....	140
Tabla 5-12. Resultados obtenidos por cada participante en EC3-PREG	140
Tabla 5-13. Valores estadísticos de los resultados obtenidos en EC3-PREG	141
Tabla 5-14. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC3-PREG	142
Tabla 5-15. Resultados de la prueba de los rangos señalados de Wilcoxon en EC3-PREG por categoría excepto composición/agregación.....	142
Tabla 5-16. Resultados obtenidos por cada participante en EC4-POST.....	143
Tabla 5-17. Valores estadísticos de los resultados obtenidos en EC4-POST.....	144
Tabla 5-18. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC4-POST	144
Tabla 5-19. Resultados de la prueba de los rangos señalados de Wilcoxon en EC4-POST por categoría excepto composición/agregación.....	145
Tabla 5-20. Resultados de la prueba de los rangos señalados de Wilcoxon en composición/agregación en EC4-POST	145
Tabla 5-21. Resultados obtenidos por cada participante en EC5-POST.....	146
Tabla 5-22. Valores estadísticos de los resultados obtenidos en EC5-POST.....	147
Tabla 5-23. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC5-POST	147
Tabla 5-24. Resultados de la prueba de los rangos señalados de Wilcoxon en EC5-POST por categoría.....	148
Tabla 5-25. Cuestionarios procesados por cada experimento controlado	148
Tabla 5-26. Resultados obtenidos en cada experimento para evaluación de las reglas de conversión a ficheros	152
Tabla 6-1. Tipos de estudios realizados para evaluar UCPD	157
Tabla 6-2. Preguntas por cada tipo de constructor del MAM para el estudio EI-PREG..	160
Tabla 6-3. Respuestas obtenidas sobre las variables del MAM para EI-PREG	161
Tabla 6-4. Análisis estadístico descriptivo sobre la facilidad de uso percibida para EI-PREG.....	162
Tabla 6-5. Respuestas obtenidas para las preguntas de utilidad para EI-PREG.....	163
Tabla 6-6. Análisis estadístico descriptivo sobre la utilidad percibida para EI-PREG....	163
Tabla 6-7. Análisis estadístico descriptivo sobre la intención de uso para EI-PREG	164
Tabla 6-8. Tareas realizadas por los profesionales en EC1-PROF	169
Tabla 6-9. Resultados obtenidos en el cuestionario 1 del estudio EC1-PROF	172
Tabla 6-10. Resultados obtenidos en el cuestionario 2 del estudio EC1-PROF	172
Tabla 6-11. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para el estudio EC1-PROF	174
Tabla 6-12. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para el estudio EC1-PROF	174

Tabla 6-13. Resultados de la prueba de los rangos señalados de Wilcoxon de técnicas ad hoc vs. UCPD para EC1-PROF.....	175
Tabla 6-14. Resultados obtenidos en el cuestionario 3 para el estudio EC1-PROF.....	176
Tabla 6-15. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para el estudio EC1-PROF.....	177
Tabla 6-16. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para EC1-PROF.....	178
Tabla 6-17. Resultados de la prueba de los rango señalados de Wilcoxon de la técnicas ad hoc vs. UCPD (caso de estudio 1) para EC1-PROF.....	178
Tabla 6-18. Análisis estadístico descriptivo para los resultados de la aplicación de UCPD en diferentes casos de estudio para EC1-PROF.....	180
Tabla 6-19. Resultados de la prueba de los rango señalados de Wilcoxon para UCPD empleando los casos de estudio 1 y 2 para EC1-PROF.....	181
Tabla 6-20. Resultados obtenidos en el cuestionario 2 para los que seleccionaron otra secuencia para el estudio EC1-PROF.....	182
Tabla 6-21. Análisis estadístico descriptivo para el cuestionario 2 de todos los participantes del estudio EC1-PROF.....	182
Tabla 6-22. Resultados de la prueba de Mann-Whitney para el cuestionario 2 para el estudio EC1-PROF.....	183
Tabla 6-23. Tareas realizadas por los participantes en EC2-POST.....	185
Tabla 6-24. Resultados obtenidos en el cuestionario 1 para el estudio EC2-POST.....	186
Tabla 6-25. Resultados obtenidos en el cuestionario 2 para el estudio EC2-POST.....	186
Tabla 6-26. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para EC2-POST.....	188
Tabla 6-27. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para el estudio EC2-POST.....	188
Tabla 6-28. Resultados de la prueba de los rangos señalados de Wilcoxon de la técnica ad hoc vs. UCPD para el estudio EC2-POST.....	189
Tabla 6-29. Resultados obtenidos en el cuestionario 1 para EC3-PREG.....	190
Tabla 6-30. Resultados obtenidos en el cuestionario 2 para EC3-PREG.....	190
Tabla 6-31. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para EC3-PREG.....	192
Tabla 6-32. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para EC3-PREG.....	192
Tabla 6-33. Resultados de la prueba de los rangos señalados de Wilcoxon de la técnica ad hoc vs. UCPD para EC3-PREG.....	193
Tabla 6-34. Preguntas por cada constructor del MAM empleadas en EC1-PROF y EC2-POST.....	198
Tabla 6-35. Resultados sobre facilidad de uso percibida con profesionales, muestras de EC1-PROF y EC2-POST.....	198
Tabla 6-36. Resultados sobre utilidad percibida con profesionales.....	199
Tabla 6-37. Resultados sobre intención de uso con profesionales.....	200

Tabla 6-38. Resultados sobre facilidad de uso percibida en EC3-PREG.....	201
Tabla 6-39. Resultados sobre utilidad percibida en EC3-PREG	201
Tabla 6-40. Resultados sobre intención de uso en EC3-PREG.....	202
Tabla 6-41. Reglas para determinar la fortaleza de la relación entre dos variables	204
Tabla 6-42. Funciones de enlace por distribución de variable dependiente	205
Tabla 6-43. Información sobre el ajuste de modelos para UP vs. IU en EI-PREG.....	207
Tabla 6-44. Bondad de ajuste para UP vs. IU en EI-PREG	208
Tabla 6-45. Estimación de los parámetros UP vs IU en EI-PREG	208
Tabla 6-46. Prueba de líneas paralelas para UP vs. IU para EI-PREG.....	209
Tabla 6-47. Pseudo R ² calculado para UP vs. IU para EI-PREG	209
Tabla 6-48. Información sobre el ajuste de modelos para UP vs. IU en EC3-PREG.....	211
Tabla 6-49. Bondad de ajuste para UP vs. IU en EC3-PREG.....	211
Tabla 6-50. Estimación de parámetros para UP vs. IU en EC3-PREG	212
Tabla 6-51. Prueba de líneas paralelas para UP vs. IU en EC3-PREG	212
Tabla 6-52. Pseudo R ² calculado para UP vs. IU para EC3-PREG.....	213
Tabla 6-53. Información sobre el ajuste de modelos para FUP vs. UP en EC1-PROF	215
Tabla 6-54. Bondad de ajuste para FUP vs. UP en EC1-PROF	216
Tabla 6-55. Estimación de parámetros para FUP vs. UP en EC1-PROF	216
Tabla 6-56. Prueba de líneas paralelas para FUP vs. UP en EC1-PROF	217
Tabla 6-57. Pseudo R ² calculado para FUP vs. UP en EC1-PROF	217
Tabla 6-58. Información sobre el ajuste de modelos para FUP vs. UP en EC2-POST	218
Tabla 6-59. Bondad de ajuste para FUP vs. UP en EC2-POST.....	219
Tabla 6-60. Modelo de regresión ordinal para FUP vs. UP en EC2-POST	219
Tabla 6-61. Prueba de líneas paralelas para FUP vs. UP para EC2-POST	220
Tabla 6-62. Pseudo R ² calculado para FUP vs. UP en EC2-POST	220
Tabla 6-63. Información sobre el ajuste de modelos para FUP-UP vs. IU en E1-PROF	222
Tabla 6-64. Bondad de ajuste para FUP-UP vs. IU en E1-PROF.....	222
Tabla 6-65. Estimaciones de los parámetros para UP- FUP vs IU para EC1-PROF.....	223
Tabla 6-66. Prueba de líneas paralelas para FUP – PU vs IU en EC1-PROF	224
Tabla 6-67. Pseudo R ² calculado para FUP – PU vs IU en EC1-PROF.....	224
Tabla 6-68. Información sobre el ajuste de modelos para FUP-UP vs. IU en EC2-POST	225
Tabla 6-69. Bondad de ajuste para FUP-UP vs. IU en EC2-POST	226
Tabla 6-70. Estimaciones de los parámetros para UP- FUP vs IU para EC2-POST	226
Tabla 6-71. Prueba de líneas paralelas para FUP – UP vs IU para EC2-POST.....	227
Tabla 6-72. Pseudo R ² calculado para FUP – UP vs IU en EC2-POST	227
Tabla 7-1. Empleo de Incremental-FP en los proyectos de alumnos por semestre	232

Tabla 7-2. Conocimientos y experiencias de los estudiantes al inicio del semestre	233
Tabla 7-3. Temas de proyectos por semestre.....	234
Tabla 7-4. Software utilizado en los proyectos.....	235
Tabla 7-5. Esfuerzo real y PFSA para cada incremento.....	236
Tabla 7-6. Productividad y esfuerzos estimados para el segundo incremento	239
Tabla 7-7. Productividad y esfuerzos estimados para el tercer incremento	240
Tabla 7-8. Porcentaje de uso de PFSA de 1er y 2do incremento en el 3er incremento .	240
Tabla 7-9. Comparación entre Incremental-FP y estimación global para los dos primeros incrementos	241
Tabla 7-10. Comparación entre Incremental-FP y estimación global para el segundo incremento	242
Tabla 7-11. Comparación entre Incremental-FP y puntos de función ajustados.....	244
Tabla 7-12. Productividad real para los proyectos del semestre 2004-2	245

Índice de Figuras

Figura 1-1. Pirámide de los sistemas de información (tomado de [Kendall, 2005])	3
Figura 1-2. Modelo en cascada (tomado de [Pfleeger, 2002])	5
Figura 1-3. Representación de modelo incremental (adaptado de [Pfleeger, 2002])	6
Figura 1-4. Representación de modelo iterativo (adaptado de [Cockburn, 2008])	7
Figura 1-5. Actividades de Tupuy y técnicas que incluye	14
Figura 2-1. Cantidad de artículos seleccionados por base de datos –2(a)	30
Figura 2-2. Cantidad de artículos seleccionados por base de datos –2(b)	34
Figura 2-3. Cantidad de artículos seleccionados por base de datos –2(c)	37
Figura 2-4. Ejemplos de relaciones entre clases	42
Figura 2-5. Ejemplo de subtipo y de herencia para empleado	46
Figura 2-6. Ejemplo de clase asociación	48
Figura 3-1. Comparación entre lo planificado y la ejecución de un proyecto en cascada (tomado de [Jaaksi, 1999])	77
Figura 3-2. Comparación entre lo planificado y la ejecución de un proyecto incremental (tomado de [Jaaksi, 1999])	78
Figura 3-3. Esquema general de Tupuy	81
Figura 4-1. Representación gráfica de actor y caso de uso	84
Figura 4-2. Diagrama de casos de uso para un sistema de biblioteca	85
Figura 4-3. Ejemplo de diagrama de caso de uso para el sistema de un banco con <i>include</i>	87
Figura 4-4. Ejemplo de diagrama de caso de uso con <i>extend</i>	87
Figura 4-5. Diagrama de casos de uso con relación <i>generalization</i> entre casos de uso ...	88
Figura 4-6. Ejemplo de caso de uso incluido no abstracto	89
Figura 4-7. Resumen de los patrones de Coad	91
Figura 4-8. Patrones de transacciones de Coad en UML con ejemplos	92
Figura 4-9. Patrones de plan de Coad en UML	93
Figura 4-10. Resumen de patrones de plan de Coad en UML	93
Figura 4-11. Patrones de agregación de Coad en UML	94
Figura 4-12. Pantalla del software de Cocomo II que calcula el EAF	97
Figura 4-13. Actividades de Tupuy y técnicas que incluye	99
Figura 4-14. Diagrama de la regla 1a de ficheros	101
Figura 4-15. Diagrama de las reglas 1b y 1c de ficheros	101
Figura 4-16. Diagrama de la regla 1d de ficheros	102
Figura 4-17. Diagrama de la regla 5 de ficheros	103

Figura 4-18. Ejemplo de la regla 7 con dependencias.....	105
Figura 4-19. Ejemplo fichero de Puntos de función en UML.....	105
Figura 4-20. Ejemplos de diagrama de casos de uso para un mismo tipo de sistema de información.....	106
Figura 4-21. Casos de uso y transacciones asociadas	108
Figura 4-22. Pantalla de StarUML y Tupux en el que se muestra la transacción Registrar venta y sus ficheros asociados.....	109
Figura 4-23. Pantalla para seleccionar DET de la transacción Registrar venta.....	110
Figura 4-24. Ejemplo de UCPD	112
Figura 4-25. Ejemplo de la Regla 1 de UCPD	113
Figura 4-26. Ejemplo de la Regla 2 de UCPD	113
Figura 4-27. Ejemplo de diagrama de UCPD	114
Figura 4-28. Diagrama simplificado del ejemplo de la Figura 4-27	114
Figura 4-29. Actividades de Incremental-FP.....	115
Figura 4-30. Diagramas de casos de uso y de clases para el ejemplo de aplicación de Tupuy.....	119
Figura 4-31. Clases y ficheros de Puntos de función relacionados del ejemplo.....	120
Figura 4-32. Casos de uso y transacciones de PF asociados del ejemplo	121
Figura 4-33. Diagrama UCPD del ejemplo	123
Figura 5-1. Vista parcial del documento que contenía los casos con los diagramas E-R	131
Figura 5-2. Diseño de los experimentos controlados para la conversión a ficheros de UML ₂ FP	132
Figura 5-3. Resultados de la pregunta relacionada a cuál de las técnicas es más clara de aplicar	149
Figura 6-1. MAM propuesto por Moody	156
Figura 6-2. Pregunta sobre secuencia de construcción aplicada en EI-PREG	159
Figura 6-3. Preguntas asociadas al MAM empleadas en el cuestionario del estudio EI-PREG.....	160
Figura 6-4. Ejemplo de caso de uso incluido en los casos de estudio EC1-PROF.....	167
Figura 6-5. Pregunta ejemplo del primer caso de estudio EC1-PROF	168
Figura 6-6. Diseño del estudio EC1-PROF	168
Figura 6-7. Ejemplo de diagrama de precedencias con dos casos de uso	170
Figura 6-8. Secuencia de construcción seleccionada por los profesionales del estudio EC1-PROF.....	171
Figura 6-9. Diferencias entre la aplicación de técnicas ad hoc (sin UCPD) y de UCPD con casos de estudio diferentes para EC1-PROF	173
Figura 6-10. Diferencias entre la aplicación de técnicas ad hoc y de UCPD con el mismo caso de estudio para EC1-PROF.....	176

Figura 6-11. Diagrama de cajas con las diferencias entre la aplicación de UCPD con casos de estudio diferentes para EC1-PROF	179
Figura 6-12. Diseño del segundo experimento controlado EC2-POST	185
Figura 6-13. Diferencias entre la aplicación de técnicas ad hoc y de UCPD para EC2-POST	187
Figura 6-14. Diferencias entre la aplicación de técnicas ad hoc y de UCPD con diferentes casos de estudio para EC2-POST.....	191
Figura 6-15. Vista parcial del cuestionario empleado para los alumnos de pregrado en EI-PREG.....	197
Figura 6-16. Vista parcial del cuestionario empleado para los profesionales en EC1-PROF	197
Figura 6-17. Vista parcial del cuestionario empleado en el experimento controlado en alumnos de pregrado EC3-PREG.....	200
Figura 6-18. Fortaleza de las relaciones del MAM para estudiantes de pregrado.....	206
Figura 6-19. Distribución de valores de IU en EI-PREG.....	207
Figura 6-20. Distribución de valores de IU en EC3-PREG.....	210
Figura 6-21. Fortaleza de las relaciones del MAM para profesionales	214
Figura 6-22. Distribución de valores de UP en EC1- PROF	215
Figura 6-23. Distribución de valores de UP en EC2-POST.....	218
Figura 6-24. Distribución de valores de IU en EC1-PROF.....	221
Figura 6-25. Distribución de valores de IU en EC2-POST	225
Figura 7-1. Ejemplo de hoja en Excel para el llenado de horas trabajadas.....	235

<h1>Introducción</h1>	Capítulo
	1

El presente trabajo de investigación se enmarca dentro de lo que se conoce como Gestión de Proyectos de Desarrollo de Software. Desde esta perspectiva, se plantea el problema de la gestión de un proyecto software que emplea un modelo de ciclo de vida incremental.

Este capítulo se organiza en cuatro secciones. La Sección 1.1 presenta el problema de investigación, la Sección 1.2 muestra el área a la que corresponde esta investigación según el Swebok y el Pmbok, la Sección 1.3 detalla los objetivos de investigación planteados y la aproximación a la solución que se va a desarrollar y, finalmente, la Sección 1.4 muestra la estructura del presente documento de tesis.

1.1 Problema de Investigación

La creciente complejidad de los desarrollos de software que provocó la denominada “crisis del software” se ha tratado de abordar mediante el planteamiento de nuevos métodos, metodologías, técnicas y paradigmas para minimizar su impacto. El alcance de dichas propuestas no se limita exclusivamente a actividades relacionadas con el desarrollo en sí de los sistemas, sino que abarca también las actividades de gestión de los mismos. Una de estas actividades es la de la estimación de los proyectos software.

En esta sección, inicialmente en la Subsección 1.1.1, se mostrarán los conceptos asociados al contexto del problema de investigación, con el objetivo que se pueda luego comprender la relevancia del problema de investigación que se detalla en la Subsección 1.1.2.

1.1.1 El contexto del problema de investigación

Antes de explicar la relevancia del problema de investigación, es importante definir algunos conceptos asociados a la problemática encontrada. El problema de investigación se enmarca dentro del desarrollo de sistemas de información, la estimación del esfuerzo basada en la técnica de puntos de función [Ifpug, 2009] y los modelos de ciclo de vida incremental.

1.1.1.1 Los sistemas de información

Un sistema de información, según Piattini et. al [Piattini, 2004], es un conjunto formal de procesos que, operando sobre una colección de datos

estructurada según las necesidades de la empresa, recopilan, elaboran y distribuye la información (o parte de ella) necesaria para las operaciones de dicha empresa y para las actividades de dirección y control correspondientes (decisiones), para desempeñar su actividad de acuerdo a su estrategia de negocio.

En la actualidad, el empleo de las tecnologías de la información, para la implementación de sistemas de información en las empresas está ampliamente difundido. Por ello, cuando se menciona el término “sistema de información”, las personas suelen relacionarlo a equipos de cómputo (software y hardware) que apoyan el flujo de la información de una organización. Sin embargo, los sistemas de información existen desde el inicio que se creó la primera organización humana [Piattini, 2004].

Según Kendall y Kendall [Kendall, 2005], los sistemas de información se pueden clasificar en las siguientes categorías:

- Sistemas de procesamiento de transacciones, *Transaction Processing Systems (TPS)* en inglés;
- Sistemas de automatización de la oficina, *Office Automation Systems (OAS)* en inglés;
- Sistemas de trabajo del conocimiento, *Knowledge Work Systems (KWS)* en inglés;
- Sistemas de información gerencial, *Management Information Systems (MIS)* en inglés;
- Sistemas de apoyo a la toma de decisiones, *Decision Support Systems (DSS)* en inglés;
- Sistemas expertos e inteligencia artificial, *Expert Systems (ES) and Artificial Intelligence (AI)* en inglés;
- Sistemas de apoyo a la toma de decisiones de grupo y Sistemas de trabajo corporativo apoyados por computadora, *Group Decision Support Systems (GDSS)* y *Computer-Supported Collaborative Work Systems* en inglés;
- Sistemas de apoyo a ejecutivos, *Executive Support Systems (ESS)* en inglés.

La Figura 1-1 muestra los diferentes sistemas de información que podrían desarrollarse para una organización. En la figura, estos sistemas se representan en una pirámide que se conoce en muchas publicaciones de los años ochenta y noventa como “pirámide los sistemas de información” (para un ejemplo de ello, se puede consultar a [Leek, 1997]). En esta pirámide se indica que los Sistemas de procesamiento de transacciones (TPS) apoyan al nivel operativo, o más bajo,

de la organización. En cambio, los ESS, GDSS y CSCWS soportan el nivel estratégico, o más alto, apoyando en la toma de decisiones.

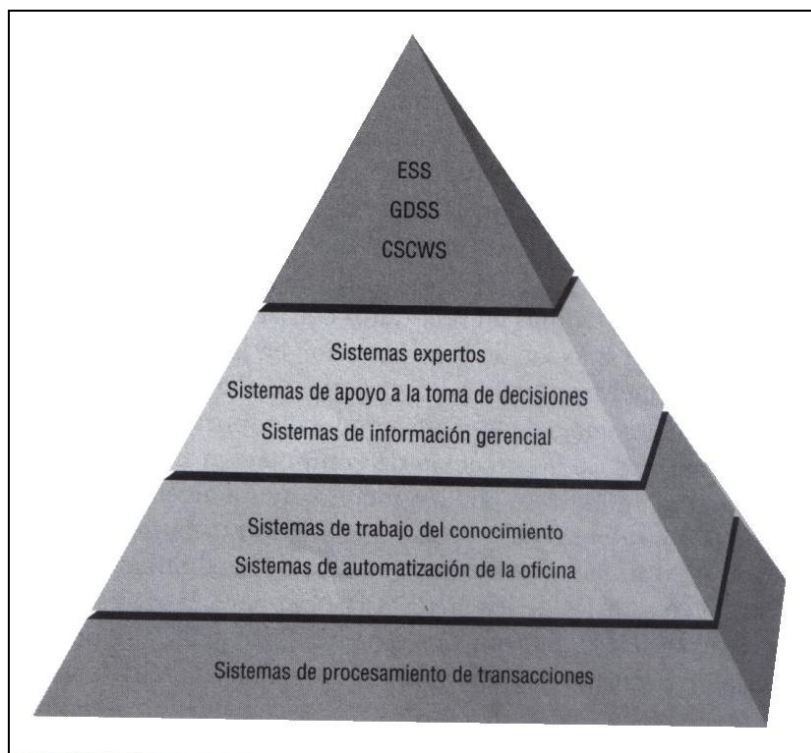


Figura 1-1. Pirámide de los sistemas de información (tomado de [Kendall, 2005])

Los Sistemas de procesamiento de transacciones, *Transaction Processing Systems (TPS)* en inglés, son sistemas de información automatizados o computarizados creados para procesar grandes cantidades de datos relacionados a transacciones rutinarias de la organización, como son los inventarios o el procesamiento de nóminas; es decir, gestiona la información referente a las transacciones producidas en una empresa u organización. Debido a que estos sistemas son empleados por el personal operativo de una organización (nivel más bajo de la pirámide organizacional) son utilizados por un gran porcentaje de empleados de las empresas. Además, la información registrada por estos sistemas es utilizada por los sistemas de las capas superiores de la pirámide; por lo que estos últimos dependen de los TPS.

1.1.1.2 Puntos de función

La técnica de Puntos de función [Ifpug, 2009] fue introducida por Albrecht [Albrecht, 1979] y su propósito es medir el software cualificando la

funcionalidad que proporciona externamente, basándose en el diseño lógico del sistema. Los objetivos de los Puntos de función (PF) son los siguientes:

- medir lo que el usuario pide y lo que el usuario recibe;
- medir independientemente de la tecnología utilizada en la implantación del sistema;
- proporcionar una métrica de tamaño que dé soporte al análisis de la calidad y la productividad;
- proporcionar un medio para la estimación del software;
- proporcionar un factor de normalización para la comparación de distintos software.

El análisis de PF se desarrolla considerando cinco parámetros básicos externos de un sistema: *External Input (EI)*, Entrada externa en castellano; *External Output (EO)*, Salida externa en castellano; *External Inquiry (EQ)*, Consulta externa en castellano; *Internal Logic File (ILF)*, Fichero lógico interno en castellano; y *External Interface File (EIF)*, Fichero de interfaz externo en castellano. Con estos parámetros, se determinan los puntos de función sin ajustar (PFSA). A este valor, se le aplica un factor de ajuste obtenido en base a unas valoraciones subjetivas sobre la aplicación y su entorno; es decir, las características generales del sistema.

Se ha probado la efectividad de la técnica de PF para medir el tamaño de sistemas de información del tipo Sistemas de procesamiento de transacciones (TPS) [Curtis, 1993]. Aunque ninguno de los manuales de la técnica [Ifpug, 1999] [Ifpug, 2004] [Ifpug, 2009] indican que solo se le debería emplear para realizar la medición de software asociado a TPS, los ejemplos que se incluyen sólo corresponden a este tipo de sistemas. Sin embargo, según lo indicado en la subsección anterior (subsección 1.1.1.1), en una organización, la mayoría de empleados utiliza un TPS y la información registrada en este tipo de sistemas es empleado por aquellos sistemas que se encuentra en los niveles superiores de la llamada “pirámide de sistemas”. Estos últimos son utilizados por los funcionarios encargados de tomar decisiones estratégicas de la organización.

Esto quiere decir que la técnica de PF es muy útil para la gestión de proyectos de desarrollo de un conjunto importante de sistemas de información en una organización. Además, su empleo es ampliamente difundido, incluso se indica como ejemplo de prácticas para la estimación de proyectos de software en el *CMMI for development* [SEI, 2010].

1.1.1.3 Los modelos de ciclo de vida

Un proceso es una serie de pasos que involucran actividades, restricciones y recursos que producen una determinada salida. Según Pfleeger [Pfleeger, 2002], cuando el proceso implica la construcción de algún producto, se suele referir al proceso como “ciclo de vida”. Es por ello que al proceso de desarrollo

de software se le denomina “ciclo de vida del software”; porque describe la vida de un producto de software desde que se concibe hasta que se implementa, entrega, opera, mantiene y retira.

Son muchos los modelos de proceso o modelos de ciclo de vida que se han descrito en la literatura sobre Ingeniería de Software. Sin embargo, los más empleados y conocidos son los siguientes: cascada, incremental e iterativo.

El modelo de ciclo de vida en cascada [Royce, 1970], en inglés *waterfall*, fue uno de los primeros modelos que se propusieron para el desarrollo de software. En este modelo, las etapas se representan cayendo en una cascada, desde una etapa hasta la siguiente; tal y como se puede ver en la Figura 1-2. En este modelo, una etapa de desarrollo debe completarse antes de dar comienzo a la siguiente. Cabe resaltar que la técnica de PF [Ifpug, 2009] puede ser empleada para medir el tamaño del software y luego estimar el esfuerzo requerido en un proyecto que sigue el modelo de ciclo de vida en cascada.

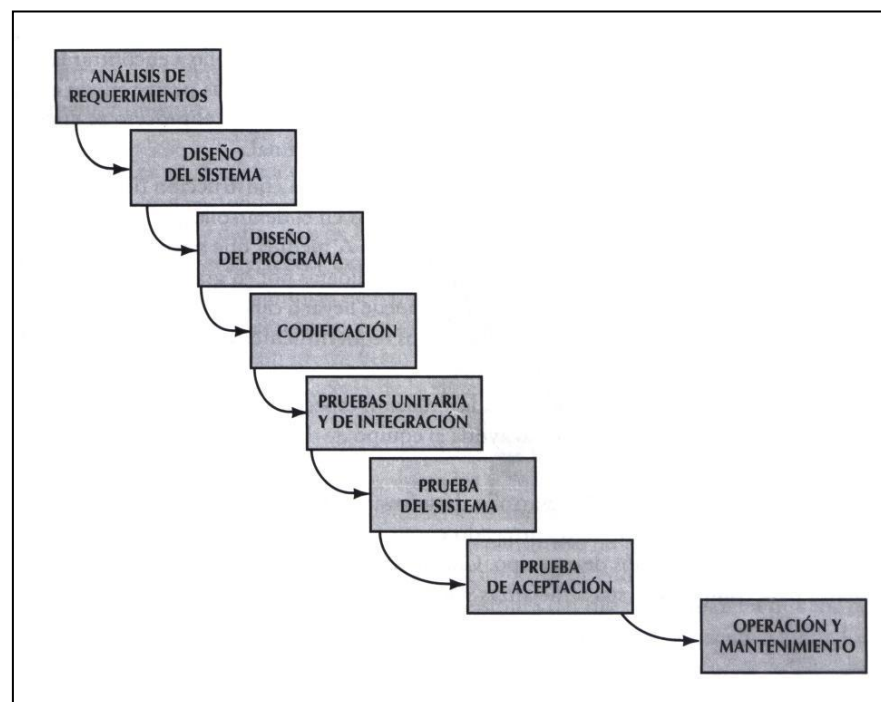


Figura 1-2. Modelo en cascada (tomado de [Pfleeger, 2002])

Sin embargo, este modelo de ciclo de vida ha recibido muchas críticas. Según Jaaksi et. al [Jaaksi, 1999], el modelo ha tenido “mala reputación”; porque su estructura rígida no toma en cuenta la naturaleza del software y asume que todo puede hacerse perfectamente en un principio, y, en muchos casos, esto no es cierto.

La norma ISO/IEC TR 15271 [ISO, 1998] indica que existen algunos factores de riesgo que hay que tomar en cuenta antes de poder considerar el empleo de un modelo del ciclo de vida en cascada para un proyecto. Algunos de estos factores de riesgo son los siguientes: que existan requisitos que no sean bien entendidos, que el sistema sea demasiado grande para hacerlo completamente al mismo tiempo, que existan cambios rápidos en la tecnología a emplear y que se produzcan cambios rápidos en los requisitos.

Cabe resaltar que este modelo de ciclo de vida en cascada también posee algunas ventajas [Jaaksi, 1999]. Una de ellas es que el modelo es fácil de entender por gente que no está relacionada a la gestión de proyectos. También, cuando se sigue este modelo, los métodos y procesos son fáciles de documentar. Además, un proyecto que emplea este modelo es fácil de gestionar y muchos jefes de proyecto prefieren diseñar y monitorear proyectos siguiendo su estructura que de por sí es rígida.

Pfleeger [Pfleeger, 2002] menciona que en los inicios del desarrollo de software, los clientes no tenían inconvenientes en esperar largo tiempo hasta que los sistemas estuviesen listos. En algunas ocasiones, podía ocurrir que pasaran años desde el momento en que se escribían los requisitos hasta que el software era entregado. Actualmente, esto ha cambiado y los negocios actuales ya no pueden tolerar tales demoras. Una de las formas para reducir el tiempo de construcción de software es el empleo del desarrollo por fases, también conocido como desarrollo por incrementos o por iteraciones.

El modelo de ciclo de vida incremental, también llamado “mejora proyectada del producto” [ISO, 1998], emplea el concepto de desarrollar por fases. Según este modelo, la construcción del software empieza con un conjunto de requisitos dados, el desarrollo se realiza en una secuencia de fases y en cada una de ellas se van añadiendo nuevas funcionalidades. Por ejemplo, como se puede observar en la Figura 1-3; la primera versión del software incorpora una parte de los requisitos, representado por un rectángulo horizontal; la segunda versión añade algunos más, representado por un rectángulo vertical; y, finalmente, la última versión incorpora los requisitos faltantes para completar el software, representado por otro rectángulo horizontal.

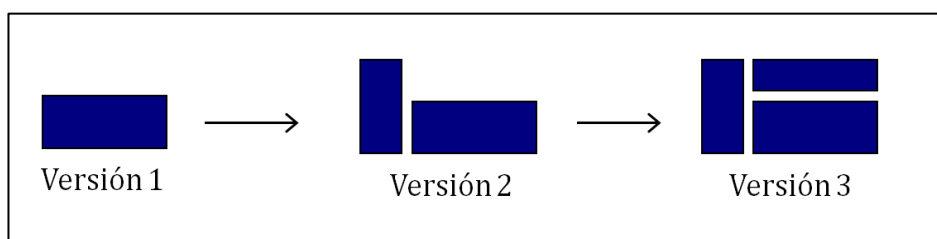


Figura 1-3. Representación de modelo incremental (adaptado de [Pfleeger, 2002])

El desarrollo iterativo [Pfleeger, 2002] entrega un sistema completo desde el principio, y luego cambia la funcionalidad para cada subsistema con cada nueva versión. Cockburn [Cockburn, 2008] presenta un ejemplo muy ilustrativo, basado en la pintura de La Gioconda de Leonardo da Vinci, para entender lo que significa un desarrollo iterativo. La Figura 1-4 muestra una adaptación del ejemplo presentado por Cockburn.

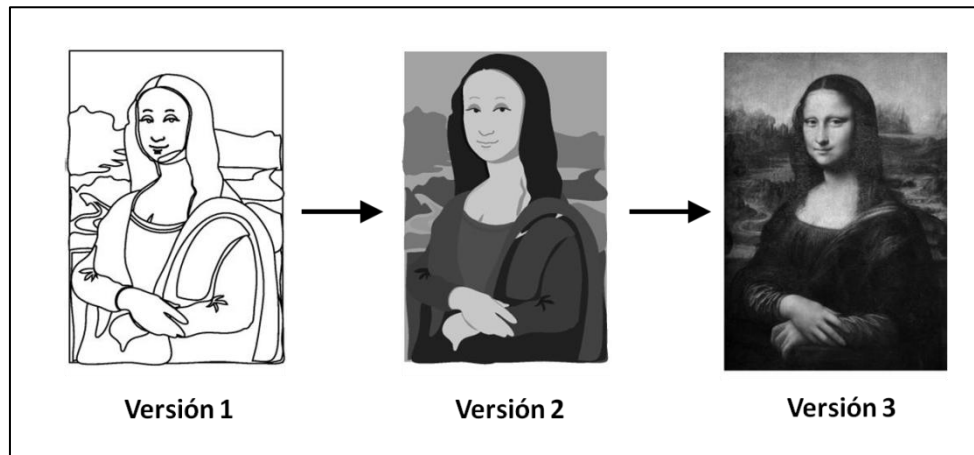


Figura 1-4. Representación de modelo iterativo (adaptado de [Cockburn, 2008])

Como se puede observar en la Figura 1-4, las tres versiones de la pintura de La Gioconda están completas. Sin embargo, una versión es el refinamiento de la anterior. Por ejemplo, la versión 2 es un refinamiento o mejora de la versión 1 y la versión 3 que es la versión final es el refinamiento o mejora de la versión 3.

Si bien el modelo de ciclo de vida incremental y el modelo de ciclo de vida iterativo son modelos de desarrollos por fases, como se puede notar en las explicaciones dadas en los párrafos anteriores, estos no son iguales. Cabe resaltar que algunos autores como Jaaksi [Jaaksi, 1999], hablan de un modelo de ciclo de vida “iterativo e incremental”, pero están haciendo referencia al modelo de ciclo de vida incremental y no al modelo de ciclo de vida iterativo.

En resumen, Cockburn [Cockburn, 2008], da algunas precisiones sobre los ciclos de vida incremental e iterativo:

- La palabra *increment*, incremento en castellano, significa *add onto*, “añadir” en castellano.
- La palabra *iterate*, iterar en castellano, significa *re-do*, “volver a hacer” en castellano.

- El desarrollo iterativo, lamentablemente, ha llegado a significar tanto incremental o iterativo de manera indiscriminada. Sin embargo, persiguen objetivos distintos y deben ser gestionados de manera diferente.
- El desarrollo incremental da la oportunidad para mejorar el proceso de desarrollo, así como ajustar los requisitos.
- El desarrollo iterativo ayuda a mejorar la calidad del producto, porque se hace una revisión de lo implementado y se actualiza para que sea cada vez mejor.

Cabe resaltar que el desarrollo de software incremental es una parte fundamental en los enfoques ágiles, tan en boga en la actualidad. Además, según Sommerville [Sommerville, 2011], el desarrollo incremental refleja la forma en que se resuelven los problemas; porque rara vez se trabaja por adelantado una solución completa de un problema y más bien se avanza en una serie de pasos hacia una solución y se retrocede cuando se detecta que se han cometido errores. Por ello, el empleo del modelo de ciclo de vida incremental es altamente aconsejable en la actualidad.

1.1.2 Relevancia del problema de investigación y dificultades que plantea

La utilización de los casos de uso para la especificación de los requisitos en los procesos de Ingeniería de Software fue propuesta por Ivar Jacobson y publicada en su libro “Object Oriented Software Engineering” [Jacobson, 1992]. Actualmente, su empleo se ha extendido aún más, debido a su inclusión en UML [OMG, 2008], por lo que su uso en el desarrollo de software orientado a objetos se ha vuelto altamente recomendable, si no obligatorio. Aunque las metodologías ágiles sugieren el empleo de las “historias de usuario”, *user stories* en inglés, en vez de los casos de uso; existen autores como Ambler [Ambler, 2004] y Leffingwell [Leffingwell, 2011] que sugieren emplear casos de uso en enfoques ágiles.

La técnica de PF [Ifpug, 2009], si bien se postula como independiente de la tecnología final de desarrollo, fue concebida para su aplicación en sistemas de información basados en el paradigma estructurado [Yourdon, 1989]. Sin embargo, David Longstreet, en uno de sus artículos [Longstreet, 2001], señala que el análisis de PF se puede aplicar de manera sencilla en casos de uso mejorando la calidad de los documentos de requisitos y, a la vez, mejorando la estimación del proyecto de software.

Aunque existen algunas propuestas para realizar el cálculo de PF con diagramas en UML [OMG, 2008] adicionalmente al de casos de uso, ninguna de ellas cumple las reglas definidas en la última documentación de la técnica de Puntos de función [Ifpug, 2004] [Ifpug, 2009].

La predicción del esfuerzo es muy diferente para un proyecto de software que emplee un modelo de ciclo de vida incremental que cuando se quiere

estimar a nivel de todo un proyecto de manera global [Abrahamsson, 2007]. La técnica de PF fue pensada para estimar el esfuerzo de todo un proyecto siguiendo un ciclo de vida en cascada de Royce [Royce, 1970]. Lamentablemente, existen muy pocas propuestas que permitan realizar la estimación del esfuerzo en proyectos de software que sigan modelos de ciclo de vida iterativo o incremental. Además, tal como se destalla más adelante, ninguna de ellas indica claramente cómo distribuir la cantidad de PF entre los diferentes incrementos.

Por tanto, esta investigación aborda el problema determinar la cantidad de PF de un proyecto desarrollado bajo el paradigma orientado a objetos. Además, se incluye la adaptación de la técnica de PF para que pueda ser utilizada para la estimación del esfuerzo en proyectos que siguen un modelo de ciclo de vida incremental.

1.2 Área de Investigación

La guía Swebok [Swebok, 2004] organiza los temas reconocidos y relacionados a la Ingeniería del Software en diez áreas del conocimiento: Requisitos del software, Diseño del software, Construcción del software, Pruebas del software, Mantenimiento del software, Gestión de la configuración del software, Gestión de la Ingeniería del Software, Proceso de la Ingeniería del Software, Herramientas y métodos de la Ingeniería del Software y Calidad del software.

De las áreas anteriormente citadas, la Gestión de la Ingeniería del Software es organizada en las siguientes subáreas: Iniciación y definición del alcance, Planificación del proyecto software, Ejecución del proyecto de software, Revisión y evaluación, Cierre y Medición de la Ingeniería de Software.

La investigación realizada en el presente trabajo se focaliza en algunas de las tareas de la subárea de Planificación del proyecto de software. La Tabla 1-1 presenta las tareas correspondientes a cada subárea de la Gestión de la Ingeniería de Software y cuáles de ellas han sido consideradas en la investigación. En la tabla, se ha indicado con un guión, “-”, aquellas tareas que no han sido consideradas en la investigación. Aquellas que sí están estrechamente relacionadas con la investigación, se ha colocado “Sí”; y aquellas que tienen alguna relación, se ha indicado con la palabra “Relacionado”.

Tabla 1-1. Tareas del Swebok consideradas en la investigación

Subárea	Tarea	Considerada en la Investigación
Iniciación y Definición de Alcance	Determinación y negociación de requerimientos	-
	Análisis de Factibilidad	-
	Proceso para la revisión de requisitos	-
Planificación del proyecto software	Planificación del proceso	Sí
	Determinación de entregables	Sí
	Estimación del esfuerzo, costo y calendario	Sí
	Asignación de recursos	-
	Gestión del riesgo	-
	Gestión de la calidad	-
	Gestión del plan	-
Ejecución del Proyecto de Software	Implementación de planes	-
	Gestión del contrato de proveedores	-
	Implementación del proceso de medición	Relacionado
	Proceso de monitoreo	Sí
	Proceso de control	-
	Reportes	-
Revisión y evaluación	Determinación de satisfacción de requerimientos	-
	Revisión y evaluación de desempeño	Relacionado
Cierre	Determinación del cierre	-
	Actividades de cierre	-
Medición en Ingeniería del Software	Establecimiento y mantenimiento del compromiso de medición	Relacionado
	Planificación del proceso de medición	Relacionado
	Realización del proceso de medición	Relacionado
	Evaluación del proceso	Relacionado

Como se puede observar en la Tabla 1-1, las tareas que forman parte de la investigación son las siguientes:

- **Planificación del proceso**, que consiste en la selección del modelo de ciclo de vida del software apropiado (por ejemplo, en cascada, espiral, etc.) y la adaptación y la implementación de los procesos apropiados del ciclo de vida del software que se llevan a cabo a la luz del alcance particular y de los requisitos que se han especificado del proyecto. También, se emplean métodos y herramientas apropiadas, como el *Work Breakdown Structure* [PMI, 2008], para descomponer el proyecto en tareas, entradas y salidas de cada tarea.

Para esta tarea, parte de nuestra investigación se enfoca **en el orden en qué se deben implementar los requisitos de software**, los cuales son representados por casos de uso [Jacobson, 1992], **siguiendo un modelo de ciclo de vida incremental**. Esto influirá en la definición y orden de las tareas que se realizarán en un proyecto.

- **Determinación de entregables**, que consiste la especificación y caracterización del producto o los productos de cada tarea (por ejemplo, la arquitectura o el diseño). Se evalúan las oportunidades para la reutilización de componentes de software de desarrollos anteriores o para emplear componentes que se puedan comprar.

Sobre esta tarea, nuestra investigación se focaliza en la determinación de **qué parte del software**, también conocido como incremento, **se entregará en un determinado momento del proyecto**. Esto influenciará en la definición de qué entregable se completará durante el desarrollo del proyecto.

- **Estimación del esfuerzo, costo y calendario**, que consiste en determinar el rango esperado de esfuerzo requerido para cada tarea. Para ello, se debería utilizar un modelo de estimación calibrado con la información histórica del esfuerzo requerido en proyectos previos o el empleo de otros métodos como el juicio de expertos. Se establecen dependencias entre las tareas, se identifican mediante métodos adecuados los cuellos de botella actuales y potenciales. Los requerimientos de recursos (personal, herramientas) se traducen en estimaciones de costo.

Para esta tarea, nuestra investigación se centra en **la estimación del esfuerzo requerido en horas-persona para la construcción de cada incremento de software**. Aunque, la investigación no toma en cuenta los requerimientos de recursos, como el caso del personal necesario para el proyecto, la estimación del esfuerzo apoya en la decisión de cuántas personas se necesitarían para una fase o para todo el proyecto.

- **Proceso de monitoreo**, que consiste en la evaluación continua y a intervalos predeterminados de la adhesión a los distintos planes. Los entregables se evalúan en términos de sus características; por ejemplo, a través de revisiones y auditorías. El esfuerzo empleado, el cumplimiento de los calendarios y de los costos hasta la fecha son revisados e investigados. Se evalúa la desviación entre los valores reales y estimados. También, se examina el empleo de los recursos. Se revisa el perfil de riesgo del proyecto y se evalúa el cumplimiento de los requisitos de calidad.

En cuanto a esta tarea, nuestra investigación se enfoca en **evaluar el esfuerzo que fue requerido para la implementación de un incremento**, con el objetivo de **determinar el esfuerzo que se necesita para construir los siguientes incrementos**. Con esto se puede determinar la desviación actual del proyecto y así poder estimar el esfuerzo requerido para la entrega de todo el software.

En cuanto a las tareas que estarían relacionadas al área de investigación, las técnicas propuestas permitirán apoyar en la revisión y evaluación de desempeño; porque se podría determinar **si la eficiencia del equipo es la adecuada** o ha habido **una variación significativa en cuanto al desempeño previo del proyecto**.

También, se puede observar que todas las tareas asociadas a “medición” se han indicado como relacionadas. Esto se debe a que no se puede desligar la estimación del esfuerzo a técnicas de medición o a lo que se conoce como **métricas de software**.

Debido a que el área de investigación corresponde a la gestión de un proyecto, no se puede dejar de lado lo que especifica el Project Management Institute en el Pmbok [PMI, 2008]. La Tabla 1-2 presenta las áreas y la descripción de las áreas del conocimiento propuestas por el Pmbok para la gestión de un proyecto de software. También, de manera similar a la Tabla 1-1, en la tercera columna de la tabla se indica qué área ha sido considerada en esta investigación. En la tabla, se ha marcado con una guión, “-”, a aquellas tareas que no han sido consideradas y con la palabra sí a aquellas que están estrechamente relacionadas con la investigación.

Tabla 1-2. Áreas del conocimiento del Pmbok consideradas en la investigación

Área del conocimiento del Pmbok	Descripción	Considerada en la Investigación
Gestión de la integración	Procesos y actividades que integran varios elementos de la gestión de proyectos, los que son identificados, definidos, combinados, unificados y coordinados dentro de los grupos de la Gestión de Proyectos. Se incluyen procesos como el Desarrollo del plan del proyecto	-
Gestión del alcance	Procesos involucrados en determinar que el proyecto incluye todo el trabajo requerido, y solo el trabajo requerido, para completar de manera exitosa el proyecto. Incluye la Planificación del alcance, Definición del alcance, Creación del <i>Work Breakdown Structure (WBS)</i> , Verificación del alcance y Control del alcance.	Sí
Gestión del tiempo	Procesos que conciernen a completar a tiempo el proyecto. Consiste en los procesos de Definición de actividades, Definición de la secuencia de actividades, Estimación de los recursos para las actividades, Estimación de la duración de las actividades, Calendarización del desarrollo y Control del calendario	Sí
Gestión del costo	Procesos involucrados en planificación, estimación, definición del presupuesto y control de costos; de tal manera que el proyecto cumple con el presupuesto aprobado. Comprende los procesos de Estimación de costos, Presupuesto de costos y Control de costos	-
Gestión de la calidad	Procesos involucrados en el aseguramiento que el proyecto satisfacerá con sus objetivos	-
Gestión de recursos humanos	Procesos para organizar y gestionar el equipo del proyecto	-
Gestión de Comunicaciones	Procesos relativos a la generación oportuna y apropiada, recopilación, difusión, almacenamiento y disposición final de la información del proyecto	-

Área del conocimiento del Pmbok	Descripción	Considerada en la Investigación
Gestión del riesgo	Procesos relacionados a la dirección de la gestión de riesgos en un proyecto.	-
Gestión de aprovisionamiento	Procesos relacionados a la compra o adquisición de productos, servicios o resultados. También, incluye los procesos para la gestión de contratos.	-

Como se puede observar en la tabla anterior, según el Pmbok el área de investigación se centra en sólo dos de sus áreas del conocimiento: Gestión del alcance, porque parte de nuestra investigación se enfoca **en el orden en qué se deben implementar los requisitos de software** y esto está relacionado al alcance del proyecto; y a Gestión del tiempo, porque nuestra investigación se enfoca en **la estimación del esfuerzo requerido en horas-persona para la construcción de cada incremento de software**.

1.3 Objetivos y Aproximación de la Solución

El objetivo de esta investigación es la creación de un conjunto de técnicas que ayude en la planificación y estimación basada en Puntos de función para determinar el esfuerzo en proyectos de software desarrollados bajo el paradigma orientado a objetos y modelos de ciclo de vida incremental. Para alcanzar este objetivo general, será necesario lograr tres subobjetivos:

- Definir una técnica que permita realizar el cálculo de Puntos de función empleando modelos orientados a objetos.
- Definir una técnica que permita determinar la priorización de casos de uso para su construcción.
- Definir una técnica basada en Puntos de función que permita calcular el esfuerzo requerido en cada incremento para un modelo de ciclo de vida incremental.

A este conjunto de técnicas se le ha denominado Tupuy que significa “medir” o “pesar” en lengua quechua, lengua de los incas. La Figura 1-5 muestra las técnicas que propone Tupuy en nubes y también se indican las actividades que comprende cada técnica.

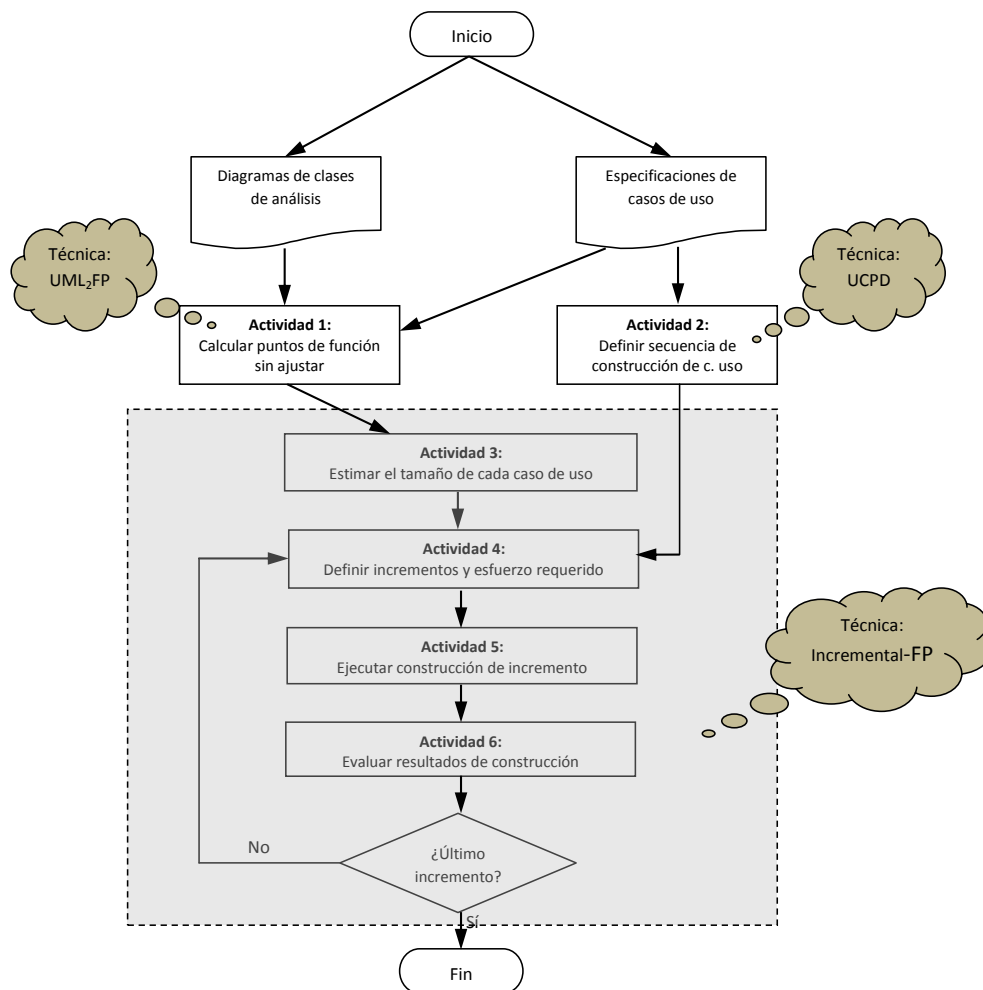


Figura 1-5. Actividades de Tupuy y técnicas que incluye

Como se puede observar en la Figura 1-5, Tupuy está conformada por tres técnicas: UML₂FP, *Use Case Precedence Diagram* (UCPD) e Incremental-FP. UCPD está conformada por la Actividad 1; UML₂FP, por la Actividad 2; e Incremental-FP por las actividades 3, 4, 5 y 6.

La técnica UML₂FP permite realizar el cálculo de PFSA (actividad 1 en la Figura 1-5). Para ello, emplea el diagrama de clases de análisis [Jaaksi, 1999], el cual también es conocido en otras metodologías como modelo de dominio [Larman, 1999]. También, utiliza las especificaciones de caso de uso.

Para la **técnica UCPD** se emplea la especificación de los casos de uso para definir la secuencia de construcción de casos de uso (actividad 2 en la Figura 1-5).

Finalmente, la **técnica Incremental-FP** utiliza los resultados obtenidos de emplear las técnicas UCPD y UML₂FP, la cual comprende la ejecución de cuatro actividades. Con estas actividades se definen los incrementos a construir, se determinan qué casos de uso se van a desarrollar en cada incremento y se estima el esfuerzo que se requiere para concluir cada incremento.

1.4 Estructura del Documento

La organización de este trabajo de tesis es la siguiente:

- En el capítulo 2, se presenta una revisión del estado de la cuestión. Para la revisión del estado actual del área de investigación, se ha hecho una revisión sistemática, siguiendo las indicaciones propuestas por Kitchenham [Kitchenham, 2007].
- En el capítulo 3, se presenta el planteamiento del problema. En este capítulo se describe el problema, se plantean las hipótesis de trabajo y se describe, de manera muy general, la solución propuesta.
- En el capítulo 4, se detalla la solución propuesta. Como se indicó en la sección anterior (Sección 1.3), esta solución está conformada por un grupo de técnicas a la cual se le ha denominado TUPUY.
- En el capítulo 5, se evalúa el modelo de conversión de ficheros de UML₂FP mediante experimentos controlados con profesionales, estudiantes de pregrado y estudiantes de posgrado.
- En el capítulo 6, se muestra la evaluación de UCPD. La evaluación se realizó mediante experimentos controlados con profesionales, estudiantes de posgrado y estudiantes de pregrado. También, se incluyen evaluaciones sobre la percepción subjetiva (facilidad de uso, utilidad e intención de uso) que tuvieron los participantes de los estudios sobre UCPD. Finalmente, se hace una evaluación de las relaciones que propone el MAM [Moody, 2001], el cual se utilizó como base para determinar si la percepción de los participantes a los estudios sobre UCPD fue positiva.
- En el capítulo 7, se muestra la evaluación de Tupuy y principalmente la evaluación de Incremental-FP. La evaluación se ha realizado con la información obtenida de proyectos con alumnos de pregrado.
- En el capítulo 8, se exponen las conclusiones de la tesis y sus aportaciones, tomando como referencia los objetivos planteados. También, se comentan las líneas de trabajo futuro.

- También, se incluyen varios anexos, que contienen lo siguiente:
 - Anexo A: incluyen las cadenas de búsqueda que se utilizaron en cada una de las bases de datos para la revisión sistemática. La revisión sistemática se explica en el capítulo de estado de la cuestión.
 - Anexo B: muestra mayor información sobre el ejemplo de aplicación de Tupuy que se incluye en el capítulo 4.
 - Anexo C: muestra las evidencias de los experimentos controlados realizados para la validación de UML₂FP: formatos empleados y formularios completados por los participantes. La explicación de esta evaluación se incluye en el capítulo 5.
 - Anexo D: muestra todas las evidencias de los experimentos controlados realizados para la validación de UCPD: formatos empleados y formularios completados por los participantes. La explicación de esta evaluación se incluye en el capítulo 6.
 - Anexo E: muestra las evidencias e información complementaria de la evaluación de Incremental-FP y Tupuy mediante la información de proyectos con alumnos de pregrado. La explicación de esta evaluación se incluye en el capítulo 7.

Estado de la Cuestión	Capítulo
	2

La revisión de literatura para encontrar y analizar las publicaciones relacionadas al área en el cual se quiere investigar es usualmente el paso inicial para comenzar un trabajo de investigación. En este capítulo se presentará una revisión de las aproximaciones existentes sobre el área de investigación. Para ello, se ha aplicado un proceso de “Revisión Sistemática”, conocido en inglés como *Systematic Review*.

Inicialmente, se definió un protocolo para encontrar las publicaciones sobre técnicas que emplearan modelos de ciclos de vida incremental, Puntos de función y paradigma orientado a objetos; sin embargo, no se encontraron artículos sobre el tema. Por ello, se tuvo que redefinir la estrategia de búsqueda, de tal manera que se pueda realizar una búsqueda por cada subpregunta de investigación previamente definida y que estaban relacionadas a lo siguiente: técnicas basadas en Puntos de función para modelos orientados a objetos, técnicas para definir la prioridad de construcción de casos de uso y técnicas para determinar el esfuerzo requerido para cada incremento en ciclos de vida incremental.

El capítulo se estructura como sigue: la sección 2.1 da una visión general de lo que es una revisión sistemática en Ingeniería del Software, la sección 2.2 presenta la definición de las preguntas de investigación, la sección 2.3 muestra la estrategia de búsqueda y selección inicial, la sección 2.4 detalla la redefinición de la estrategia de búsqueda y selección debido a que no se encontraron resultados en la búsqueda inicial, la sección 2.5 incluye una revisión de las técnicas basadas en Puntos de función para modelos orientados a objetos, la sección 2.6 muestra las técnicas para determinar la prioridad de construcción de los casos de uso que se encontraron en el proceso de búsqueda, la sección 2.7 muestra las aproximaciones existentes para calcular el esfuerzo requerido en cada incremento, la sección 2.8 presenta la discusión sobre las amenazas a la validez de la revisión sistemática y, finalmente, se incluyen las conclusiones del estado de la cuestión.

2.1 Revisión Sistemática en Ingeniería de Software

El término de Revisión sistemática, cuya denominación en inglés es “*Systematic Review*”, es empleado para referirse a una metodología de investigación que se emplea para recolectar y evaluar evidencia disponible sobre un tema específico.

En otras áreas como la Medicina, este tipo de revisiones son ampliamente utilizadas y son muy conocidas. La aplicación de este tipo de metodologías es relativamente reciente en la Ingeniería de Software.

Kitchenham es una de las investigadoras en Ingeniería de Software que ha impulsado el empleo de revisiones sistemáticas; incluso, a través de un reporte técnico, muestra los pasos a seguir para dirigir este tipo de revisiones [Kitchenham, 2004] [Kitchenham, 2007]. Otra guía para realizar este tipo de trabajos es propuesta por Biolchini et. al [Biolchini, 2005].

Aunque las revisiones sistemáticas y las revisiones de literatura tradicionales están sujetas a errores sistemáticos y aleatorios, las primeras emplean métodos de revisión científicos que minimizan la parcialidad y los errores. La Tabla 2-1 muestra las diferencias entre las revisiones tradicionales y sistemáticas que presentan Dyba et al. [Dyba, 2007] y que son una adaptación de un trabajo de Mulrow y Cook [Mulrow, 1998].

Tabla 2-1. Diferencias entre revisiones tradicionales y sistemáticas

Característica	Revisión Tradicional	Revisión Sistemática
Preguntas	A menudo son amplias en alcance	A menudo se enfocan en preguntas de investigación
Cómo se identifican las investigaciones	Usualmente no se indica y son potencialmente parcializadas	Se indican las fuentes (bases de datos de búsqueda) y estrategias de búsquedas
Selección de investigaciones	Usualmente no se indica y es potencialmente parcializado	Selección basada en criterios y son uniformemente aplicadas
Apreciación	Variable	Valoración crítica y rigurosa
Síntesis	A menudo son solo resúmenes cualitativos	Síntesis cuantitativa y/o cualitativa
Inferencias	A veces basadas en evidencias	Usualmente basadas en evidencias

A diferencia de las revisiones tradicionales, las revisiones sistemáticas permiten encontrar la mejor evidencia disponible a través de métodos para identificar, valorar y sintetizar estudios relevantes de manera rigurosa en un tema de investigación en particular. Los métodos que se emplean son previamente definidos y documentados, de tal manera que otros investigadores puedan evaluarlos de manera crítica y los puedan replicar posteriormente.

2.2 Definición de Preguntas de Investigación

Según Kitchenham [Kitchenham, 2007], la definición de preguntas de investigación es la parte más importante para realizar una revisión sistemática, que permiten dirigir adecuadamente la investigación.

Para determinar las preguntas de investigación, a sugerencia de Petticrew y Roberts [Petticrew, 2006], se tiene que emplear el criterio que ellos denominan “PICOC”. El nombre de este criterio se debe a las letras iniciales de los siguientes términos en inglés: *Population* (población), *Intervention* (intervención), *Comparison* (comparación), *Outcome* (salida) y *Context* (contexto).

Como el objetivo de esta revisión sistemática no es encontrar comparaciones entre técnicas, aproximaciones, métodos o modelos; no se ha incluido el término *Comparison* del criterio PICOC. La Tabla 2-2 muestra los resultados que se obtuvieron al aplicar este criterio.

Tabla 2-2. Resultados de la aplicación del criterio PICOC

Criterio	Descripción
<i>Population</i> (población)	Desarrollo de software orientado a objetos que empleen el modelo de ciclo de vida incremental
<i>Intervention</i> (intervención)	Métodos, técnicas, aproximaciones basadas en Puntos de función (PF) para la estimación y planificación
Outcomes (salidas)	Precisión en la predicción, métricas exitosas para estimación
Context (contexto)	El contexto es amplio: académico, industria de software y todo tipo de <i>stakeholders</i> (desarrolladores, jefes de proyecto, estudiantes, investigadores, etc.). También, incluirá cualquier tipo de estudio empírico: observaciones, cuestionarios, encuestas, experimentos controlados, casos de estudio, etc.

Según la información definida en la tabla anterior, las preguntas de investigación son las siguientes:

Pregunta 1: ¿Qué evidencia existe para la estimación y planificación basada en PF para desarrollos de software orientado a objetos que empleen el modelo de ciclo de vida incremental?

Pregunta 2: ¿Qué técnicas/métodos/aproximaciones basadas en PF han sido utilizadas para estimar y planificar desarrollos de software orientado a objetos que empleen el modelo de ciclo de vida incremental?

Pregunta 2(a): ¿Qué técnicas o modelos de conversión se han definido para calcular PF empleando modelos orientados a objetos?

Pregunta 2(a-1): ¿Qué diagramas de UML emplean las técnicas?

Pregunta 2(a-2): ¿Las técnicas cumplen con lo que especifican los últimos manuales de PF?

Pregunta 2(b): ¿Qué actividades/tareas/pasos se han definido para calcular el esfuerzo requerido para cada incremento?

Pregunta 2(c): ¿Qué actividades/tareas/pasos se han definido para determinar la prioridad en la construcción de casos de uso?

Las preguntas definidas anteriormente, permitieron conducir la revisión de aproximaciones existentes en el área.

2.3 Estrategia de Búsqueda y Selección

Para poder realizar la búsqueda y selección de artículos, se tuvo que definir una estrategia a seguir. Inicialmente se tuvo que definir los términos que se emplearían para la búsqueda en las bases de datos; luego, las fases que tendría el proceso de búsqueda; y, finalmente, los criterios que se tomarían para incluir o excluir los artículos encontrados.

2.3.1 Términos de búsqueda

Tomando en cuenta la información obtenida al aplicar el criterio PICOC (ver Tabla 2-2), se definieron los términos de búsqueda que se muestran en la Tabla 2-3. Los términos que se emplearon están en inglés, dado que es el idioma ampliamente difundido en la comunidad científica internacional.

Tabla 2-3. Términos en inglés derivados del criterio PICOC

Criterio	Términos
<i>Population</i> (población)	software development, object oriented, incremental
<i>Intervention</i> (intervención)	Method/technique/approach, estimation/plan, function point
Outcomes (Salidas)	Accuracy, prediction, succesful predictors
Context (Context)	Software industry, developer, project manager, student researcher, empirical studies, observation, interview, questionnaires, surveys, experiments, case studies.

En el criterio de intervención (*intervention*), las palabras *method* y *technique* se podrían considerar como sinónimos; es por ello que se han colocado unidas por una barra inclinada ("/"). En cuanto a las palabras *estimation* y *plan*, aunque no son sinónimos, están estrechamente relacionadas; por esa razón es que también se colocó unidas por una barra inclinada ("/").

En un inicio, solo se utilizarán en las búsquedas los términos de población (*population*) y de intervención (*intervention*). Si la cantidad de resultados que se obtuvieran en las bases de datos fuera muy grande, se emplearían los términos de salida (*outcomes*) para reducir la cantidad de artículos. Como el contexto (*context*) es muy amplio, sus términos no se usarían en ninguna de las búsquedas.

La cadena de búsqueda creada a partir de los términos *Population* y *Intervention* de la Tabla 2-3 se muestran a continuación:

software development **AND** object oriented **AND**
incremental **AND** method/technique **AND** metric **AND**
estimation/plan **AND** function point

Para encontrar más términos según el criterio PICOC, se revisaron sinónimos en inglés y bibliografía relacionada al tema. La Tabla 2-4 muestra la relación de términos y la fuente de donde se obtuvo.

Tabla 2-4. Términos relacionados para *population* e *intervention*

PICOC	Término original	Término derivado	Observación y/o fuente
Population	Software development	Software process, software engineering	Términos propuestos por Petersen [Petersen, 2011]
	Software development	Software construction	RUP [IBM, 2006] propone el empleo de la palabra "construction" para la construcción de software.
	Software development	Software project	Término ampliamente utilizado para referirse a proyectos de desarrollo de software.
	Software development	System* development, software system* development, application development, system* construction, software system* construction, application construction	Ryaz [Ryaz, 2009a] propone el empleo de los siguientes sinónimos para software: systems, software systems, applications. Se han creado nuevos términos con esas palabras
	Object oriented	UML, Unified Modeling Language	Lenguaje de modelado que se asocia a la orientación a objetos [OMG, 2008].
	Object oriented	Object orientation	Términos empleados por algunos artículos en vez de <i>object oriented</i> ; por ejemplo [Zhang, 2007]

PICOC	Término original	Término derivado	Observación y/o fuente
	Incremental	Iterative, iteration, sprint, release, increment	Términos empleados por RUP [IBM, 2006] y por Scrum [Cohn, 2010]
Intervention	Method, technique, approach	Process, system, practice, procedure	Términos empleados por Ryaz [Ryaz, 2009] debido a que son sinónimos de <i>method/technique/approach</i>
	Function point	Ifpug, NESMA	Los términos corresponden a asociaciones encargadas de realizar manuales para el cálculo de Puntos de función
	Metric	Measure, measurement	Sinónimos en inglés obtenidos de diccionarios.
	Estimation/plan	Forecast*, calculat*, predict*	Los dos últimos términos se indican por sugerencia de Jorgensen [Jorgensen, 2007]. El primero corresponde a un sinónimo en inglés obtenido en diccionarios.

Al observar los términos relacionados a *software development* de la Tabla 2-3, podría ocurrir que algunos autores consideren otras combinaciones no incluidas; por ejemplo, podrían emplear en los resúmenes de sus artículos la frase “*development of software*” y no “*software development*”. Si fuera este el caso, los buscadores de las bases de datos no encontrarían este tipo de artículos, los cuales podrían ser relevantes. Por ello, se definió una nueva cadena que permita considerar todos los casos posibles:

```
((("software" OR "application*" OR "system*") AND
("development" OR "construction" OR "project*" OR
"process*" OR "engineering"))
```

Debido a que la base de datos IEEE Xplore solo permite el empleo de cinco *wildcards* como máximo, se redujeron las palabras con asterisco (*) de tal manera que no se sobrepasará este máximo permitido. Por ejemplo, “*application**” fue reemplazado por las palabras “*application*” y “*applications*”. Finalmente, todas las cadenas básicas que se emplearon para las búsquedas fueron las siguientes:

- C1. ((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" "projects" OR "process" OR "processes" OR "engineering"))
- C2. ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language")
- C3. ("function point" OR "function points" OR "Ifpug" OR "NESMA")

- C4. ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases")
- C5. ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*")
- C6. ("method" OR "technique" OR "process" OR "system" OR "practice" OR "procedure" OR "approach")
- C7. ("metric" OR "measure*")

Todas las cadenas básicas anteriores se combinaron utilizando el operador “AND”. Esto quiere decir que los artículos deberían contener por lo menos un término de cada cadena básica. La cadena que se emplearía en las búsquedas es la siguiente:

C1 AND C2 AND C3 AND C4 AND C5 AND C6 AND C7

Las cadenas básicas C6 y C7 solo se utilizarían para reducir la cantidad de resultados, si es que era muy elevado. Por lo tanto, inicialmente, se emplearían solo para las búsquedas las cadenas C1, C2, C3, C4 y C5. La cadena obtenida solo se usaría para realizar las búsquedas por títulos y resúmenes (*abstracts*, en inglés) a fin de evitar la obtención de miles de resultados innecesarios [Riaz, 2009a] [Riaz, 2009b].

2.3.2 Fases del proceso de búsqueda

El proceso de búsqueda comprendió dos fases: búsqueda primaria y búsqueda secundaria. En el caso de la búsqueda primaria, se emplearon las siguientes bases de datos, las cuales son utilizadas por muchas revisiones sistemáticas en Ing. de Software: Scopus, ISI Web of Science, Science Direct, IEEE Xplore, ACM, Computer Database y Engineering Village (Inspec y Compendex). No se utilizó para la búsqueda primaria la base de datos Scholar Google ya que no permite el empleo de cadenas de búsquedas complejas.

Aunque existen otras bases de datos que son importantes, no se utilizaron; porque por experiencias documentadas de investigadores que han realizado revisiones sistemáticas en Ingeniería de Software, éstas no encontrarían más artículos que los que se obtendrían solo con las bases de datos que se seleccionaron para esta revisión. Riaz [Riaz, 2009] sugiere no incluir IEEE Computer Society Digital, porque confirmó que IEEE Xplore provee los mismos resultados. Además, Dyba et. al [Dyba, 2007] al realizar búsquedas, determinó que los mismos artículos obtenidos con Kluwer Online, SpringerLink y Wiley Inter Science Journal Finder fueron también obtenidos con ISI Web of Science o con Compendex.

La “literatura gris”, cuyo término en inglés es *gray literature*, según el diccionario Merriam-Webster¹, es todo aquel material escrito (por ejemplo un reporte) que no es publicado comercialmente y no es por lo general accesible. Este tipo de literatura fue excluida en las búsquedas. Engström et. al [Engström, 2008] descartan este tipo de artículos por dos razones: es difícil evaluar su calidad y las búsquedas contendrían un elevado número de resultados.

La búsqueda secundaria consiste en revisar las referencias y citaciones de los artículos obtenidos en la búsqueda primaria con el fin de encontrar artículos relevantes al tema. Adicionalmente a las bases de datos empleadas para la búsqueda primaria, se empleó Scholar Google para determinar las citaciones a los artículos seleccionados. La ventaja de Scholar Google es que se obtienen mayor cantidad de citaciones que al emplear Scopus o ISI Web of Science, porque estas dos últimas se centran en indicar las citaciones de solo aquellos artículos que estas mismas bases de datos tienen indexados. También, Scholar Google incluye artículos correspondientes a “literatura gris”.

2.3.3 Criterios de inclusión y exclusión

Los criterios de inclusión y exclusión definen las pautas que se deben seguir para determinar qué artículos deben ser seleccionados y cuáles no. Para esta revisión sistemática, los criterios de inclusión y exclusión que se definieron se detallan a continuación.

Se incluirán todos los artículos que contengan algún tipo de validación. Además, sólo se incluirán los artículos que emplean la técnica de PF, sigan el paradigma orientado a objetos y modelo de ciclo de vida incremental. No se incluirán los artículos que cumplan lo siguiente:

1. Técnicas diferentes a PF. Tampoco se incluirán aquellas técnicas que se basan en PF, pero no cumplen con las reglas para realizar los cálculos y que están definidas en los manuales del Ifpug [Ifpug, 1999], [Ifpug, 2004][Ifpug, 2009].
2. Técnicas que sigan un ciclo de vida diferente al incremental.
3. Técnicas que no sigan el paradigma orientado a objetos.
4. Publicaciones que no hayan pasado por un proceso de revisión por pares. Esto quiere decir que no se incluirán prefacios, revisiones, libros, etc.
5. Artículos de conferencias que luego fueron extendidos en artículos de algún *journal* [Lucas, 2009]

¹ [http://www.merriam-webster.com/dictionary/gray literature](http://www.merriam-webster.com/dictionary/gray%20literature)

6. Artículos cuya propuesta sea poco clara o ambigua; por ejemplo que los resultados no sean soportados por una evidencia objetiva [Lucas, 2009].
7. Artículos propios relacionados a esta tesis y que hayan sido previamente publicados.

2.3.4 Selección de artículos

Para la selección de artículos, se siguieron las sugerencias de MacDonnell y Shepperd [MacDonnell, 2007]. Por ello, se revisaron los títulos y resúmenes de los artículos encontrados para determinar si deberían ser considerados como estudios primarios. Si la decisión no se podía tomar solo con la información del resumen, se tenía que leer el artículo completo.

2.4 Proceso de búsqueda y selección

El proceso de búsqueda se realizó durante el mes de mayo de 2011 y luego se aplicó nuevamente en setiembre de 2011. Para las búsquedas, la cadena mostrada en la sección anterior se tuvo que adaptar para que pueda ser ingresada en cada una de las base de datos (en el anexo A se han incluido las cadenas que se emplearon en cada base de datos).

Se pudo notar que algunas de las bases de datos ofrecen interfaces más fáciles de usar que otras. Scopus e ISI Web of Knowledge son las que proveen mejores herramientas para búsqueda y descarga de resultados; IEEE Xplore limita el uso de *wildcards*, por ejemplo "*", a un máximo de cinco y ACM Digital Library no permite descargar los resultados en otros formatos como BibTex o archivos de texto. En cuanto a Engineering Village (Inspec & Compendex), al parecer, a diferencia de otras bases de datos, no reconoce los *wildcards* si la frase se encuentra entre comillas; por ejemplo, se debería emplear ***predict* OR effort estimat* OR cost estimat**** y no ***"predict*" OR "effort estimat*" OR "cost estimat*"***.

Al aplicar la estrategia de búsqueda que se detalla en la sección anterior (Sección 2.3), solo se pudieron obtener resultados en dos bases de datos. La Tabla 2-5 muestra la relación de artículos encontrados por cada base de datos y cuántos artículos fueron seleccionados.

Tabla 2-5. Resultados de búsqueda inicial

Base de Datos	Artículos Encontrados	Artículos Seleccionados
Scopus	2	0
ISI Web of Science	1	0
Science Direct	0	0

Base de Datos	Artículos Encontrados	Artículos Seleccionados
IEEE Xplore	0	0
ACM	0	0
Computer Database	0	0
Engineering Village (EI & Compendex)	2	0

Según los criterios de inclusión y exclusión definidos, no se seleccionó ningún artículo. Pero a pesar de ello, uno de los artículos encontrados seguía un modelo de ciclo de vida iterativo; el cual, al igual que el ciclo de vida incremental, permite el desarrollo de software por partes o incrementos. El artículo en mención fue el siguiente:

- Hericko M., Zivkovic A., The size and effort estimates in iterative development, Information and Software Technology, Vol. 80, pp. 772-781, Elsevier, 2008 [Hericko, 2008].

Como se obtuvieron pocos resultados con las bases de datos y no se seleccionó ningún artículo, se tuvo que redefinir la estrategia de búsqueda y selección para poder obtener más artículos relacionados a las preguntas de investigación definidas. Además, por la experiencia que se obtuvo con esta búsqueda, se cambiaron los criterios de exclusión; de tal manera que se pudieran seleccionar también los artículos que empleen el ciclo de vida iterativo.

2.5 Redefinición de la Estrategia de Búsqueda y Nuevo Proceso de Búsqueda

Debido a los pocos resultados obtenidos al emplear la estrategia de búsqueda y selección que se muestra en la sección anterior (ver Sección 2.3), se revisaron las preguntas de investigación que previamente se habían definido y se tuvo que hacer búsquedas que permitieran responder a las subpreguntas definidas en la pregunta 2: **¿Qué técnicas/métodos/aproximaciones basadas en PF han sido utilizadas para estimar y planificar desarrollos de software orientado a objetos que empleen ciclos de vida incremental?** Las subpreguntas son las siguientes:

- **Pregunta 2(a):** ¿Qué técnicas o modelos de conversión se han definido para calcular PF empleando modelos orientados a objetos?
- **Pregunta 2(b):** ¿Qué actividades/tareas/pasos se han definido para calcular el esfuerzo requerido para cada incremento?
- **Pregunta 2(c):** ¿Qué actividades/tareas/pasos se han definido para determinara la prioridad en la construcción de casos de uso?

Además, para encontrar la mayor cantidad de artículos, se decidió emplear solo los términos relacionados a *Population* y a *Intervention*; dejando de lado los criterios *Comparison*, porque no se quiere comparar con otras técnicas; *Outcome*, porque podría ocurrir que los términos empleados en este criterio no estén incluidos en el resumen del artículo y se tendrían que incluir muchos sinónimos (p.ej: *successful predictor*, *quality*, etc.); y *Context*, porque el contexto es muy amplio (experimentos controlados, estudios cualitativos, industria, academia, etc.).

A continuación se muestran los resultados obtenidos al realizar las búsquedas para encontrar técnicas de cálculo de puntos de función en base a modelos orientados a objetos (Subsección 2.5.1), técnicas para priorizar los requisitos para la construcción de software (Subsección 2.5.2) y técnicas para calcular el esfuerzo requerido para cada incremento (Subsección 2.5.3).

2.5.1 Pregunta de investigación 2(a): Técnicas basadas en Puntos de función para modelos orientado a objetos

Siendo la pregunta de investigación definida **¿qué técnicas o modelos de conversión se han definido para calcular PF empleando modelos orientados a objetos?**, se tuvo que determinar los nuevos términos de búsqueda según el criterio PICOC. En este caso, al igual que en la estrategia inicial de búsqueda (ver subsección 2.3.1), no se incluyó el criterio de comparación (*comparison* en inglés), porque no se quiere buscar comparaciones, ni el criterio de contexto (*context* en inglés), porque es muy amplio.

Tabla 2-6. Resultados de la aplicación del criterio PICOC para la pregunta 2(a)

Criterio	Descripción	Términos
<i>Population</i> (población)	Desarrollo de software orientado a objetos	software development, object oriented
<i>Intervention</i> (intervención)	Métodos, técnicas basadas en puntos de función	Function point, Method/technique/approach

La cadena de búsqueda creada a partir de los términos *population* y *intervention* de la Tabla 2-6 se muestra a continuación:

software development **AND** object oriented **AND** function point **AND** method/technique **AND** accuracy/precision

Considerando los términos equivalentes que se detallan en la Tabla 2-3 de la sección anterior (Sección 2.3), las cadenas básicas de búsqueda fueron las siguientes:

- C1. (("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" "projects" OR "process" OR "processes" OR "engineering"))
- C2. ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language")
- C3. ("function point" OR "function points" OR "Ifpug" OR "NESMA")
- C4. ("method" OR "technique" OR "process" OR "system" OR "practice" OR "procedure" OR "approach")

Para encontrar la mayor cantidad de artículos, se prefirió no incluir la cadena básica C4. Por lo tanto, la cadena que se empleó fue la siguiente: C1 **AND** C2 **AND** C3

Los criterios de inclusión/exclusión fueron modificados para esta búsqueda. Para esta pregunta de investigación, solo se incluirían los artículos que tuvieran algún tipo de validación, aproximaciones que empleen la técnica de Puntos de función y que utilicen modelos orientados a objetos. Los criterios de exclusión para la selección de artículos fueron los siguientes:

1. Técnicas diferentes a PF o técnicas que se basan en PF, pero no cumplen con sus reglas para realizar los cálculos y que están definidas en los manuales del Ifpug [Ifpug, 1999], [Ifpug, 2004] [Ifpug, 2009].
2. Técnicas que no sigan el paradigma orientado a objetos.
3. Técnicas que tengan reglas específicas para desarrollos en plataforma Web.
4. Técnicas que consideran modelos de conversión para etapas tardías de un proyecto de desarrollo de software como diseño o programación.
5. Artículos que no hayan pasado por un proceso de revisión por pares. Esto quiere decir que no se incluirían prefacios, revisiones, etc.
6. Artículos de conferencias que luego fueron extendidos en artículos de *journals*.
7. Artículos propios relacionados a esta tesis y que hayan sido previamente publicados.

La cantidad de resultados que se obtuvieron al emplear la cadena de búsqueda en las bases de datos se muestra en la Tabla 2-7. Cabe resaltar, que el número indicado en la primera columna corresponde a la secuencia en la que se realizaron las búsquedas. La cantidad de artículos duplicados corresponde a aquellos artículos que no se encontraron en las búsquedas realizadas previamente; por ejemplo al realizar la búsqueda en Science Direct, se

encontraron 7 artículos; sin embargo, solo uno de ellos no fue encontrado ni en ISI ni en Scopus.

Tabla 2-7. Cantidad de artículos encontrados y seleccionados para la pregunta 2(a)

Nº	Base de Datos	Número Artículos Encontrados	Número de Artículos No Duplicados	Número de Artículos Seleccionados
1	Scopus	83	-	11
2	ISI Web of Science	52	13	0
3	Science Direct	7	1	0
4	IEEE Xplore	23	1	0
5	ACM	7	1	0
6	Computer Database	4	1	0
7	Engineering Village (INSPEC & Compendex)	78	1	0

Para seleccionar los artículos, se siguió la siguiente estrategia: se agruparon los estudios por autor y se descartaron aquellos artículos de conferencias que habían sido detallados posteriormente en algún artículo de *journal* y solo se consideraron estos últimos. Luego de realizar la búsqueda en las bases de datos indicadas y revisar los contenidos de los artículos, se seleccionaron las publicaciones que se muestran en la Tabla 2-8.

Tabla 2-8. Artículos seleccionados para la pregunta 2(a)

Nº	Autores	Título	Tipo de Publicación	Año
1	Fetcke T., Abran A., Nguyen T.	Mapping the OO-Jacobson approach into function point analysis [Fetcke, 1997]	Conferencia	1997
2	Antoniol G., Lokan C., Caldiera G., Fiutem R.	Function point-like measure for object-oriented software	Journal	1999
3	Antoniol G., Fiutem R., Lokan C.	Object-oriented function points: An empirical validation	Journal	2003
4	Uemura T., Kusumoto S., Inoue K.	Function-point analysis using design specifications based on the unified modelling language	Journal	2001
5	Cantone G., Pace D., Calavaro G.	Applying function point to unified modeling language: Conversion model and pilot study	Conferencia	2004
6	Harput V., Kaindl H., Kramer S.	Extending function point analysis to object-oriented requirements specifications	Conferencia	2005
7	Zivkovic A., Rozman I., Hericko M.	Automated software size estimation based on function points using UML models	Journal	2005
8	Abraham S., Poels G., Pastor O.	A functional size measurement method for object-oriented conceptual schemas: Design and evaluation issues	Journal	2006

Nº	Autores	Título	Tipo de Publicación	Año
9	Abrahao S., Poels G.	Experimental evaluation of an object-oriented function point measurement procedure	Journal	2007
10	Gupta D., Kaushal S.J., Sadiq M.	Software estimation tool based on three - layer model for software engineering metrics	Conferencia	2008
11	Lavazza L.A., Bianco V.D., Garavaglia C.	Model-based functional size measurement [Lavazza, 2008]	Conferencia	2008
12	Chamundeswari A., Babu C.	Function point size estimation for object oriented software based on use case model	Conferencia	2008

La Figura 2-1 muestra la cantidad de artículos seleccionados que fueron encontrados en cada base de datos. Como se puede observar, Scopus, ISI Web of Science y Engineering Village fueron las bases de datos que permitieron obtener la mayor cantidad de artículos que finalmente fueron seleccionados. La base de datos que obtuvo menos resultados fue Computer Database.

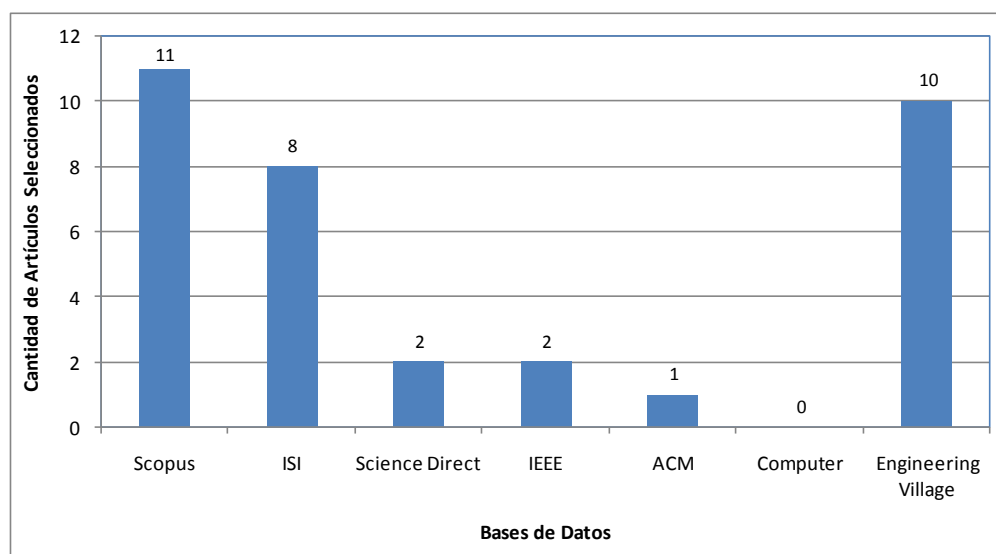


Figura 2-1. Cantidad de artículos seleccionados por base de datos –2(a)

Lo mostrado en la Figura 2-1 nos podría sugerir que solo era necesario emplear Scopus para las búsquedas, ya que solo con esta base de datos se obtuvieron todos los artículos que fueron seleccionados. Sin embargo, esto no se puede afirmar fehacientemente, ya que se necesitarían analizar los resultados obtenidos en otras búsquedas para poder comprobarlo. También, se puede observar que no se seleccionó ningún artículo de Computer Database.

En cuanto a la búsqueda secundaria, inicialmente se revisaron las referencias de los artículos seleccionados. Muchas de estas referencias

correspondían a artículos que previamente se habían encontrado en las bases de datos y que habían sido o no seleccionados. No obstante, se pudo identificar un artículo que contenía más información sobre la técnica propuesta por Lavazza [Lavazza, 2008]. Este artículo [Del Bianco, 2008] se incluyó en la relación de publicaciones a revisar.

Cuando se empleó Scholar Google para las citaciones de los artículos seleccionados, se pudo identificar publicaciones que fueron previamente encontradas con las bases de datos y que no fueron seleccionadas, o artículos que cumplían con los criterios de exclusión y/o no cumplían con los criterios de inclusión. Por lo tanto, el análisis de citaciones no permitió identificar nuevos artículos.

La sección 2.6 presenta el detalle de la revisión de los artículos seleccionados en la búsqueda primaria y secundaria para esta pregunta de investigación 2(a): ¿Qué modelos de conversión se han definido para calcular PF empleando modelos orientados a objetos?

2.5.2 Pregunta de investigación 2(b): Técnicas para calcular el esfuerzo requerido por cada incremento

Como la pregunta de investigación es **¿qué actividades/tareas/pasos se han definido para calcular el esfuerzo requerido para cada incremento?**, se definieron los nuevos términos de búsqueda según el criterio PICOC. De manera similar a la estrategia inicial de búsqueda que se detalla en la subsección 2.3.1, no se incluyeron los criterios de comparación (*comparison*, en inglés) ni el de contexto (*context*, en inglés). Tampoco se consideró el criterio de salida (*outcome*, en inglés) para poder obtener la mayor cantidad de resultados. Para este caso no se consideró que los proyectos emplearan el paradigma orientado a objetos.

Tabla 2-9. Resultados de la aplicación del criterio PICOC para la pregunta 2(b)

Criterio	Descripción	Términos
<i>Population</i> (población)	Desarrollo de software que empleen ciclo de vida incremental	software development, incremental
<i>Intervention</i> (intervención)	Métodos, técnicas basadas en Puntos de función para determinar la planificación y estimación	Method/technique, metric, estimation/plan, function points

La cadena de búsqueda creada a partir de los términos *population* y *intervention* de la Tabla 2-9 se muestra a continuación:

software development **AND** incremental **AND**
method/technique **AND** metric **AND** estimation/plan **AND**
function point

Considerando los términos equivalentes que se detallan en la Tabla 2-3 de la sección anterior (Sección 2.3), las cadenas básicas de búsqueda fueron las siguientes:

- C1. ("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" "projects" OR "process" OR "processes" OR "engineering"))
- C2. ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases")
- C3. ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*")
- C4. ("function point*" OR "Ifpug" OR "NESMA")
- C5. ("method" OR "technique" OR "process" OR "system" OR "practice" OR "procedure" OR "approach")

La cadena que se empleó para la búsqueda fue C1 **AND** C2 **AND** C3 **AND** C4 **AND** C5. Los criterios de inclusión/exclusión fueron modificados para esta búsqueda. Para esta pregunta de investigación, se incluirían los artículos que tengan algún tipo de validación, aproximaciones que empleen la técnica de PF y que sigan el modelo de ciclo de vida incremental o el iterativo (por experiencia de la estrategia inicial de búsqueda, mostrada en la Sección 2.3, se incluyen los modelos de ciclo de vida iterativo los cuales tienen elementos en común con el incremental). Los criterios de exclusión fueron los siguientes:

1. Técnicas diferentes a PF. Tampoco se incluirían aquellas técnicas que se basan en PF, pero no cumplen con sus reglas para realizar los cálculos y que están definidas en los manuales del Ifpug [Ifpug, 1999], [Ifpug, 2004] [Ifpug, 2009].
2. Técnicas que no sigan el modelo de ciclo de vida incremental ni el iterativo.
3. Estudios que no hayan pasado por un proceso de revisión por pares. Esto quiere decir que no se incluirán prefacios, revisiones, etc.
4. Artículos de conferencias que luego fueron extendidos en artículos de *journals*.
5. Artículos propios relacionados a esta tesis y que hayan sido previamente publicados.

La cantidad de resultados que se obtuvieron al aplicar la cadena de búsqueda definida en las bases de datos se detallan en la Tabla 2-10. El número indicado en la primera columna corresponde a la secuencia en la que se realizaron las búsquedas. La cantidad de estudios duplicados se debe a que en búsquedas previas en otras bases de datos no se obtuvieron esos artículo; por ejemplo, con ISI Web of Science se obtuvieron 2 artículos que no se encontraron en Scopus.

Tabla 2-10. Cantidad de artículos encontrados y seleccionados para la pregunta 2(b)

No	Base de Datos	Número Artículo Encontrados	Número de Artículos No Duplicados	Número de Artículos Seleccionados
1	Scopus	21	-	3
2	ISI Web of Science	10	2	0
3	Science Direct	2	2	0
4	IEEE Xplore	8	8	0
5	ACM	1	1	0
6	Computer Database	1	1	0
7	Engineering Village (INSPEC & Compendex)	15	15	0

Luego de realizar la búsqueda en las bases de datos indicadas y revisar los contenidos de los artículos, se seleccionaron los artículos que se muestran en la Tabla 2-11.

Tabla 2-11. Artículos seleccionados para la pregunta 2(b)

No	Autores	Título	Tipo de publicación	Año
1	Orr G., Reeves T.E.	Function point counting: One program's experience	Journal	2000
2	Kang S., Choi O., Baik J.	Model-based dynamic cost estimation and tracking method for agile software development	Conferencia	2010
3	Hericko M., Zivkovic A.	The size and effort estimates in iterative development	Journal	2007

La Figura 2-2 muestra la cantidad de artículos seleccionados que fueron encontrados en cada base de datos. Como se puede observar, Scopus fue la única base de datos que permitió encontrar los tres estudios seleccionados. Algo

similar ocurrió para la pregunta de investigación 2(a) (ver Subsección 2.5.1) y nos podría sugerir que solo sería necesario emplear Scopus para las búsquedas.

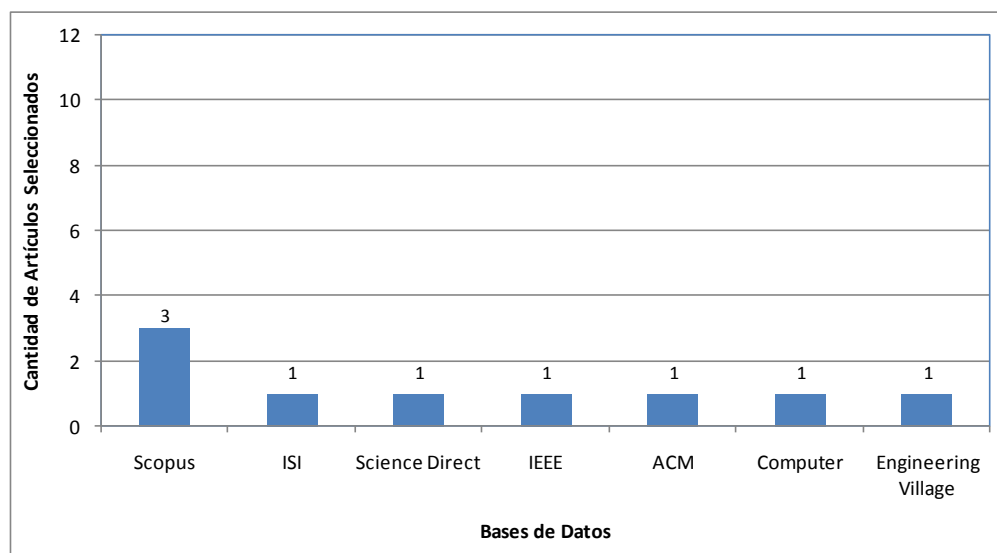


Figura 2-2. Cantidad de artículos seleccionados por base de datos –2(b)

Cuando se empleó Scholar Google para las citaciones de los artículos seleccionados, se pudo identificar publicaciones que cumplieran con los criterios de exclusión y/o no cumplieran con los criterios de inclusión. Por lo tanto, el análisis de citaciones no permitió identificar nuevos artículos. La revisión de referencias no permitió identificar más artículos que cumplieran con los criterios de inclusión.

2.5.3 Pregunta de investigación 2(c): Técnicas para priorizar o definir la secuencia de la construcción de casos de uso

Como la pregunta de investigación es **¿qué actividades/tareas/pasos se han definido para determinar la prioridad en la construcción de casos de uso?**, se revisaron los tres artículos que fueron seleccionados en la subsección anterior (Subsección 2.5.2) correspondiente a la pregunta **¿qué actividades/tareas/pasos se han definido para calcular el esfuerzo requerido para cada incremento?** Lamentablemente, ninguno de ellos indica cómo definir qué caso de uso debe construirse en cada incremento o iteración. Por ello, se tuvo que realizar una nueva búsqueda relacionada directamente a técnicas para priorizar la construcción de casos de uso. La Tabla 2-13 muestra los términos identificados para *Population* e *Intervention* del criterio PICOC para esta pregunta.

Tabla 2-12. Términos en inglés para *population* e *intervention* para la pregunta 2(c)

Criterio	Descripción	Términos
<i>Population</i> (población)	Desarrollo de software	Software development
<i>Intervention</i> (intervención)	Métodos, técnicas o aproximaciones para determinar la priorización de la construcción de los casos de uso	Method/technique/approach, prioritization, use case

La cadena de búsqueda general sería software development AND method/technique/approach AND priorit*/sequenc* AND use case*. Adicionalmente, se buscaron términos asociados a los incluidos en la cadena general. La Tabla 2-13 muestra estos términos adicionales y cuál es la fuente u observación considerada para incluirlos.

Tabla 2-13. Términos en inglés derivados para *population* e *intervention* para la pregunta 2(c)

Término original	Término derivado	Fuente/observación
Use case	User story	User story es una técnica equivalente a casos de uso y que es sugerida por las metodologías ágiles [Cohn, 2010]
Use case	Increment, sprint, release	Términos que se emplean para la construcción de software cuando se siguen ciclos de vida evolutivos o incrementales. Sprint es un término que se emplea en Scrum [Cohn, 2010].
Priorit*	Precedence, precede	Término empleado por Iconix para determinar que caso de uso precede a otro. Podría ser tomado como sinónimo de prioridad [Rosenberg, 1999]
Software development	Requirements engineering	Este término es muy empleado para identificar los procesos de requisitos de desarrollo de software. Algunos artículos indican este término y dan por asumido que corresponde al desarrollo de software

Considerando los términos equivalentes que se detallan en la Tabla 2-3 de la sección anterior (Sección 2.3) y de la Tabla 2-13, las cadenas básicas de búsqueda fueron las siguientes:

C1. (("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR

```
"construction" OR "project" OR "projects" OR  
"process" OR "processes" OR "engineering"))
```

C2. ("priorit*" OR "precede*")

C3. ("use case" OR "use cases" OR "user story" OR "user stories" OR "sprint" OR "sprints" OR "release" OR "releases")

C4. ("method" OR "technique" OR "process" OR "system" OR "practice" OR "procedure" OR "approach")

Sin embargo, al aplicar las cadenas de búsqueda C1, C2 y C3 en Scopus, se obtuvieron más de 13,199 resultados. Luego, se añadió la cadena C4 y los resultados se redujeron a 12, 813. Por ello, se tuvo que cambiar la cadena C1 de tal manera que solo se asociara a software. La nueva cadena C1 fue la siguiente:

```
C1. ("software development" OR "software construction" OR  
"software project" OR "software projects" OR  
"software process" OR "software processes" OR  
"software engineering" OR "requirement* engineering")
```

Los criterios de inclusión/exclusión fueron modificados para esta búsqueda. Para esta pregunta de investigación, se incluirían los estudios que tengan algún tipo de validación y que correspondan a proyectos de desarrollo de software. Los criterios de exclusión fueron los siguientes:

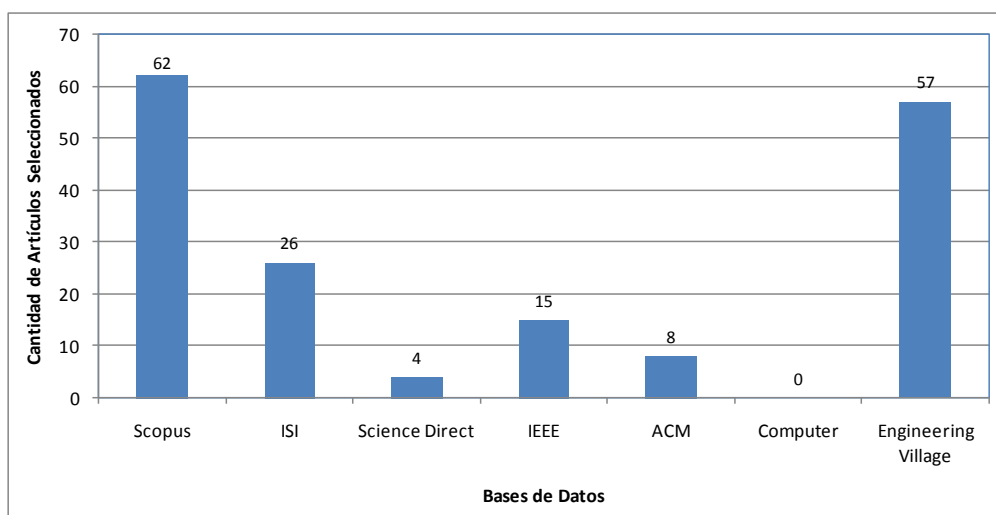
1. Artículos que no hayan pasado por un proceso de revisión por pares. Esto quiere decir que no se incluirían prefacios, revisiones, etc.
2. Artículos de conferencias que luego fueron extendidos en artículos en *journals*.
3. Artículos propios relacionados a esta tesis y que hayan sido previamente publicados.

La cantidad de resultados que se obtuvieron al aplicar la cadena de búsqueda definida en las bases de datos se detallan en la Tabla 2-1. El número indicado en la primera columna corresponde a la secuencia en la que se realizaron las búsquedas. La cantidad de estudios duplicados se debe a esta secuencia; por ejemplo, en IEEE Xplore se obtuvieron 21 artículos, 4 de los cuales no se habían encontrado con Science Direct, ISI Web of Science ni Scopus.

Tabla 2-14. Cantidad de artículos encontrados y seleccionados para la pregunta 2(c)

Nº	Base de Datos	Número Estudios Encontrados	Número de Estudios No Duplicados	Número de Estudios Seleccionados
1	Scopus	113	-	62
2	ISI Web of Science	32	6	6
3	Science Direct	5	0	0
4	IEEE Xplore	21	4	3
5	ACM	9	2	1
6	Computer Database	0	0	0
7	Engineering Village (INSPEC & Compendex)	95	1	0

Para seleccionar los artículos, se siguió la misma estrategia que la empleada para la pregunta 2(a) (ver subsección 2.5.1): se agruparon los estudios por autor y se descartaron aquellos artículos de conferencias que habían sido detallados posteriormente en un *journal* y solo se consideraron estos últimos. Finalmente se seleccionaron 72 artículos (en el anexo A se muestra la lista de estos artículos). La Figura 2-3 presenta la cantidad de artículos seleccionados que fueron encontrados en cada base de datos.

**Figura 2-3.** Cantidad de artículos seleccionados por base de datos –2(c)

Como se puede observar en la Figura 2-3, Scopus fue la base de datos que permitió encontrar la mayor cantidad de resultados que finalmente fueron seleccionados. Esto difiere con lo ocurrido en las búsquedas anteriores (ver

subsecciones 2.5.1 y 2.5.2), en la que Scopus era la única base de datos que permitió encontrar todos los estudios seleccionados. Debido a estos resultados, no se puede afirmar que se debería utilizar únicamente Scopus. También, a diferencia de las búsquedas anteriores, para esta pregunta de investigación, sí se obtuvieron publicaciones en IEEE Xplore y ACM que no se habían encontrado en Scopus o ISI Web of Science.

2.6 Técnicas para Calcular Puntos de Función en Modelos Orientados a Objetos

Luego de revisar los artículos seleccionados del proceso de búsqueda, se le asignó un código a cada técnica, de manera que sea más fácil su identificación durante toda la revisión. Por ejemplo, la primera técnica tiene el código “Ta1”; porque la letra “T” significa que es una Técnica, la letra “a” viene de la codificación de la pregunta de investigación relacionada que es 2a y el número final es un correlativo.

La primera columna de la Tabla 2-15 muestra el código asignado, la segunda columna presenta la información de las publicaciones asociadas a cada técnica y la última columna indica la versión del manual de Puntos de función empleado. Los manuales de Puntos de función son conocidos como Ifpug CPM (“CPM” es por las siglas en inglés de *Counting Practices Manual*).

Tabla 2-15. Técnicas identificadas para Puntos de función y modelos orientados a objetos

Técnica	Publicación				Versión de IPUFG CPM
	Autores	Título	Tipo	Año	
Ta1	Fetcke T., Abran A., Nguyen T.	Mapping the OO-Jacobson approach into function point analysis [Fetcke, 1997]	Conferencia	1997	4.0
Ta2	Antoniol G., Lokan C., Caldiera G., Fiutem R.	Function point-like measure for object-oriented software [Antoniol, 1999]	Journal	1999	4.0
	Antoniol G., Fiutem R., Lokan C.	Object-oriented function points: An empirical validation [Antoniol, 2003]	Journal	2003	
Ta3	Uemura T., Kusumoto S., Inoue K.	Function-point analysis using design specifications based on the unified modelling language [Uemura, 2001]	Journal	2001	4.1
Ta4	Cantone G., Pace D., Calavaro G.	Applying function point to unified modeling language: Conversion model and pilot study [Cantone, 2004]	Conferencia	2004	4.1.1
Ta5	Harput V., Kaindle H., Kramer S.	Extending function point analysis to object-oriented requirements specifications 38 [Harput, 2005]	Conferencia	2005	4.1.1
Ta6	Zivkovic A., Rozman I., Hericko M.	Automated software size estimation based on function points using UML models 101 [Zivkovic, 2005]	Journal	2005	4.1

Técnica	Publicación				Versión de IPUFG CPM
	Autores	Título	Tipo	Año	
Ta7	Abrahao S., Poels G., Pastor O.	A functional size measurement method for object-oriented conceptual schemas: Design and evaluation issues 2 [Abrahao, 2006]	Journal	2006	4.1
	Abrahao S., Poels G.	Experimental evaluation of an object-oriented function point measurement procedure 3 [Abrahao, 2007]	Journal	2007	
Ta8	Lavazza L.A., Bianco V.D., Garavaglia C.	Model-based functional size measurement [Lavazza, 2008]	Conferencia	2008	4.2
	Del Bianco, V., Gentile, C., Lavazza, L.	An Evaluation of Function Point Counting Based on Measurement-Oriented Models [Del Bianco, 2008]	Conferencia	2008	4.2
Ta9	Chamundeswari A., Babu C.	Function point size estimation for object oriented software based on use case model [Chamundeswari, 2008]	Conferencia	2008	4.1

Como las últimas versiones del manual de Puntos de función son 4.2.1 y 4.3.1, se puede observar en la Tabla 2-15 que la mayoría de las técnicas han utilizado versiones antiguas del manual. Solo la técnica Ta8 de Lavazza et. al ha empleado una de las últimas versiones del Ifpug CPM.

Para el cálculo de puntos de función, se consideran como parámetros básicos para determinar el tamaño funcional de un software lo siguiente:

- **Ficheros lógicos internos**, *Internal Logic Files* (ILF) en inglés. Son grupos de datos lógicamente relacionados identificables por el usuario, o información de control contenida dentro de los límites de la aplicación.
- **Ficheros de interfaz externos**, *External Interface File* (EIF) en inglés. Son grupos de datos lógicamente relacionados identificables por el usuario, o de control utilizada en la aplicación pero mantenida por otra.
- **Entradas externas**, *External Input* (EI) en inglés. Son datos o información de control que procede de fuera de los límites de la aplicación
- **Salidas externas**, *External Output* (EO) en inglés. Son datos o información de control que sale fuera de los límites de la aplicación mediante un proceso lógico.
- **Consultas externas**, *External Inquiry* en inglés, (EQ). Son procesos elementales que obtienen una combinación de entrada/salida como resultado de una recuperación de datos, pero no mantienen ningún ILF.

Los ILF y EIF son denominados por la técnica como *Data Functions*, funciones de datos en castellano, y los EI, EO y EQ son denominados como *Transactional Functions*, funciones transaccionales en castellano. Para mayor facilidad en la lectura de este documento, a los ILF y EIF se les denominará a partir de ahora como “ficheros” y a los EI, EO y EQ como “transacciones”.

Una vez identificados todos los parámetros (ILF, EIF, EI, EQ, EO), se determinan la cantidad de Puntos de función sin ajustar a cada uno de ellos. A este valor se le aplicará a continuación un factor de ajuste a partir de una serie de valoraciones subjetivas que tendrán que ver con el entorno en el que se desarrollará el proyecto, así como con su aplicación posterior; es decir, se trata de características generales del sistema.

A continuación se realizará la revisión y comparación de lo que propone cada una de las técnicas. La Subsección 2.6.1 presenta las propuestas de las técnicas a nivel de ficheros, la Subsección 2.6.2 muestra las propuestas para las transacciones y, finalmente, la Subsección 2.6.3 presenta las conclusiones de toda la revisión realizada para Puntos de función.

2.6.1 Ficheros de Puntos de Función

Los últimos manuales para contar PF [Ifpug, 2004] [Ifpug, 2009] definen cómo identificar los ficheros empleando diagramas entidad-relación (E-R) [Chen, 1975]. Aunque este tipo de diagramas corresponde al paradigma de desarrollo estructurado [Yourdon, 1989] comparten muchos elementos en común con un diagrama de clases del paradigma orientado a objetos [Jacobson, 1992][Rumbaugh, 1991].

Para poder determinar la cantidad de PFSA por cada fichero, sea ILF o EIF, primero se tiene que identificar la cantidad de DET y RET que tiene cada fichero. A partir de la versión 4.2 del Ifpug CPM [Ifpug, 2004], se definen reglas para identificar ficheros, DET y RET empleando diagramas E-R. La Tabla 2-16 muestra estas reglas cuando existen dos entidades A y B que se relacionan.

Tabla 2-16. Reglas para determinar ficheros en PF (tomado de [Ifpug, 2009])

Type of relationship between two entities, A and B	When this Condition Exists	Then Count as LFs with RETs and DETs as follows:
(1) : (N)	(A and B are independent)	2 LFs, 1 RET and DETs to each
1 : N	If B is entity dependent on A	1 LF, 2 RETs, sum DETs
	If B is entity independent of A	2 LFs, 1RET and DETs to each
1 : (N)	If B is entity dependent on A	1 LF, 2 RETs, sum DETs
	If B is entity independent of A	2 LFs, 1 RET and DETs to each
(1) : N	If A is entity dependent on B	1 LF, 2 RETs, sum DETs
	If A is entity independent of B	2 LFs, 1RET and DETs to each
(1) : (1)	(A and B are independent)	2 LFs, 1RET and DETs to each
1 : 1	(A and B are dependent)	1 LF, 1 RET, sum DETs
1 : (1)	If B is entity dependent on A	1 LF, 1 or 2 RETs, sum DETs
	If B is entity independent of A	2 LFs, 1 RET and DETs to each
(N) : (M)	(A and B are independent)	2 LFs, 1 RET and DETs to each
N : M	If B is entity dependent on A	1 LF, 2 RETs, sum DETs
	If B is entity independent of A	2 LFs, 1 RET and DETs to each
N : (M)	If B is entity dependent on A	1 LF, 2 RETs, sum DETs
	If B is entity independent of A	2 LFs, 1 RET and DETs to each

Para poder determinar si dos entidades A y B son consideradas como un solo fichero o que cada entidad es un fichero, se debe revisar primero la cardinalidad que tiene la relación entre ambas entidades y que se muestra en la primera columna de la Tabla 2-16. Por ejemplo, si la cardinalidad es de cero a uno en A y de cero a muchos en B, se debe emplear la regla de la primera fila. En el caso que la cardinalidad sea de uno en A y de uno a muchos en B, se debería emplear la regla de la segunda fila.

Luego de revisar la cardinalidad de la relación entre las dos entidades y de haber seleccionado a qué fila de la Tabla 2-16 corresponde, se debe determinar si las entidades son independientes o dependientes. Una entidad es dependiente de otra, cuando ésta solo existe si la otra existe; esto quiere decir que si una entidad se destruye, la otra también debería ser destruida. Por ejemplo, en el caso que la cardinalidad entre A y B sea de “1:N” y la entidad B sea dependiente, entonces se debe escoger la segunda fila de la Tabla 2-16 y esta indica que A y B se reconocen como un solo fichero con dos RET y se suman la cantidad de atributos de A y de B como DET. Para el mismo caso, si las entidades fueran independientes, entonces A y B se reconocerían como ficheros independientes, cada una con su propio RET y su propia cantidad de DET.

La técnica de PF también especifica que en el caso de independencia entre entidades se debe añadir un DET adicional si la cardinalidad relacionada a una entidad es de muchos, la cual se considera en la Tabla 2-16 como “N” o “M”. Por ejemplo, para el caso la segunda fila de la tabla, si las entidades fueran independientes, se debería contar un DET adicional a B debido a la cardinalidad de muchos.

Las revisiones y comparaciones entre lo que propone cada técnica para los ficheros se realizará teniendo en cuenta los siguientes criterios: clases y atributos (ver acápite 2.6.1.1); relaciones de asociación, agregación y composición entre clases (ver acápite 2.6.1.2); relaciones de herencia (ver acápite 2.6.1.3); y clase asociación (ver acápite 2.6.1.4). En el acápite 2.6.1.5, se muestra un resumen de la revisión realizada de las técnicas con relación a los ficheros.

2.6.1.1 Clases y atributos

Todas las técnicas encontradas proponen emplear los diagramas de clases para poder identificar los ficheros de PF. Sin embargo, algunas de las técnicas indican que solo se deben emplear las clases de tipo entidad, mas no las clases de tipo control o interfaz (*boundary* en inglés), si es que se utiliza el diagrama de clases propuesto por Jacobson [Jacobson, 1992].

Las técnicas, además, proponen emplear los atributos de cada clase para determinar la cantidad de DET que tendría cada fichero. Sin embargo, la técnica Ta4 propuesta por Cantone et. al [Cantone, 2004] y la técnica Ta9 propuesta por Chamundeswari et. al [Chamundeswari, 2008]; adicionalmente, indica que debería contarse un RET por cada atributo complejo de cada clase; por ejemplo la referencia a un objeto. Esto último no cumpliría con lo definido en la Tabla 2-16 correspondiente al manual de Puntos de función, ya que el equivalente de

una “referencia a un objeto” es una “relación a una entidad” de un diagrama E-R y estos son considerados como DET cuando la cardinalidad es de muchos y nunca como RET.

2.6.1.2 Relaciones de asociación, agregación y composición entre clases

La Figura 2-4 muestra ejemplos de relaciones que pueden existir entre clases. Entre la clase A y B existe una relación de asociación, entre C y D una relación de agregación, y entre E y F hay una relación de composición. La agregación y composición son un tipo de asociación que se emplean cuando existe una relación todo/parte entre clases; por ejemplo: auto/rueda. Para estos casos, el rombo se muestra próximo a la clase que representa el todo. La diferencia entre la composición y agregación radica en la dependencia entre clases, ya que la composición indica que las partes solo pueden existir siempre y cuando el todo existe. Esto quiere decir que en una relación de composición si el todo se destruye, sus partes también deberían destruirse.

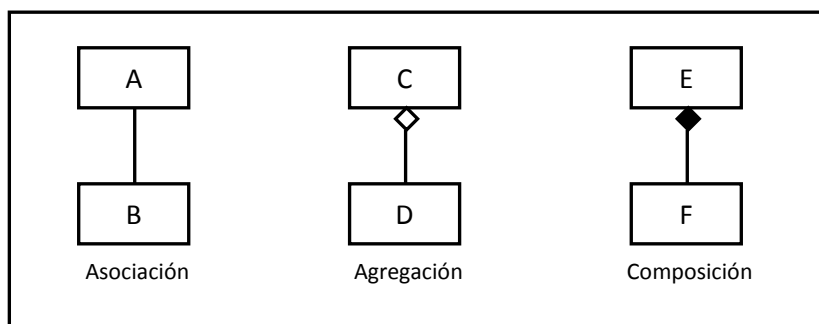


Figura 2-4. Ejemplos de relaciones entre clases

Empleando como ejemplo lo mostrado en la Figura 2-4, se identificó lo que propone cada técnica para la relación de asociación y si cumple con lo que establecen los últimos manuales de PF. La primera columna de la Tabla 2-17 muestra el código que se le ha asignado a la técnica; la segunda columna, los autores; la tercera columna, la propuesta de la técnica para la relación de asociación entre las clases A y B (ver Figura 2-4) y la última columna indica si la técnica cumple o no con los últimos manuales de PF. En esta última columna, se indica con un guión (“-”) si es que la técnica no define reglas para asociación.

Tabla 2-17. Propuestas de las técnicas para asociación

Código	Autores	Reglas para Asociación	¿Cumple con Ifpug CPM 4.2.1 y 4.3.1?
Ta1	Fetcke T., et. al	No definido	-
Ta2	Antoniol G., et. al	Si A tiene multiplicidad de 1, a B se le añade un DET. Si es mayor a 1, a B se le añade un RET.	No, porque cuando es multiplicidad de 1 no se añade DET. Además, la multiplicidad de muchos no añade RET.
Ta3	Uemura T., et. al	No definido	-
Ta4	Cantone G., et. al	Emplea propuesta de Antoniol (Ta2)	No
Ta5	Harput V., et. al	DET adicional para A y para B	No, porque solo se añade DET adicionales si la multiplicidad es de muchos
Ta6	Zivkovic A., et. al	Emplea propuesta de Antoniol (Ta2)	No
Ta7	Abrahao S., et. al	Igual que Antoniol (Ta2)	No
Ta8	Lavazza L.A., et. al	Si A y B están en un componente, contar un fichero con dos RET	No, porque las dos clases se reconocen como un fichero sólo si son dependientes.
Ta9	Chamundeswari A., et. al	Igual que Antoniol (Ta2)	No

Como se puede observar en la Tabla 2-17, muchas de las técnicas utilizan la propuesta de Antoniol et. al (Ta2) [Antoniol, 1999] o son similares a ella, pero esta propuesta no cumple con lo que definido en la técnica de PF (ver Tabla 2-16). Adicionalmente, ninguna de las propuestas toma en cuenta la dependencia entre clases y esto es importante para determinar si A y B se consideran como un fichero o dos. Esto quiere decir que ninguna de las técnicas propone reglas que cumplan con lo establecido por los últimos manuales de PF.

También, se realizó un análisis a la relación de agregación, de manera similar al que se hizo para la relación de asociación. La primera columna de la Tabla 2-18 muestra el código que se le ha asignado a la técnica; la segunda columna, los autores; la tercera columna, la propuesta de la técnica para la relación de asociación entre las clases C y D, las cuales tienen una relación de agregación (ver Figura 2-4); y la última columna indica si la técnica cumple o no con los últimos manuales de PF. En esta última columna, se indica con un guión (“-”) si es que la técnica no define reglas para agregación.

Tabla 2-18. Propuestas de las técnicas para agregación

Código	Autores	Reglas para Agregación	¿Cumple con Ifpug CPM 4.2.1 y 4.3.1?
Ta1	Fetcke T., et. al	Un fichero con 2 RET, 1 por C y otro por D	No, porque podrían ser dos ficheros, ya que se tiene que evaluar la dependencia entre clases

Código	Autores	Reglas para Agregación	¿Cumple con Ifpug CPM 4.2.1 y 4.3.1?
Ta2	Antoniol G., et. al	Dependiendo del punto de vista, C y D pueden ser ficheros separados o uno solo por los dos. Añadir un RET a C, sin importar que multiplicidad tenga.	No, porque la propuesta es muy abierta y la técnica de puntos de función de guías claras para indicar cuando es un fichero y cuando dos. Además, solo se añadiría un RET a C si es que C y D son dependientes.
Ta3	Uemura T., et. al	No definido	-
Ta4	Cantone G., et. al	Añadir un RET a C, sin importar la multiplicidad que tenga, de manera similar a la propuesta de Antoniol (Ta2). También, D se puede considerar como un fichero separado, pero esta opción no es la recomendable.	No, porque la propuesta es muy abierta y la técnica de puntos de función de guías claras para indicar cuando es un fichero y cuando dos. Además, solo se añadiría un RET a C si es que C y D son dependientes.
Ta5	Harput V., et. al	Igual que Fetcke (Ta1)	No, porque podrían ser dos ficheros, ya que se tiene que evaluar la dependencia entre clases
Ta6	Zivkovic A., et. al	Emplea propuesta de Antoniol (Ta2)	No
Ta7	Abrahao S., et. al	Mantener como ficheros separados. Añadir DET y RET según reglas de asociación.	No, porque podría ser un solo fichero. Falta evaluar la dependencia entre C y D.
Ta8	Lavazza L.A., et. al	No definido	-
Ta9	Chamundeswari A., et. al	Según la multiplicidad de la relación, se sigue regla de asociación de Antoniol (Ta2): Si C tiene multiplicidad de 1, a D se le añade un DET. Si es mayor a 1, a D se le añade un RET.	No, porque cuando es multiplicidad de 1 no se añade DET. Además, la multiplicidad de muchos no añade RET.

Según lo que se observa en la Tabla 2-18, las técnicas no toman en cuenta la evaluación de dependencias entre las entidades o clases para así poder determinar si C y D representan a un solo fichero o a dos. Por lo tanto, las técnicas no cumplen con lo establecido por los últimos manuales de Puntos de función.

La relación de composición, al igual que la agregación, representa una relación todo/parte. La diferencia se encuentra en la dependencia: en una relación de composición la existencia de la parte depende de la existencia del todo. Esto quiere decir que, según lo definido en los últimos manuales de PF, si dos clases que se relacionan mediante composición, como E y F (ver Figura 2-4) deberían ser considerados como un solo fichero con dos RET.

La Tabla 2-19 muestra el resultado del análisis de la relación de composición, de manera similar al que se hizo para las relaciones de asociación y agregación. Cabe resaltar que la relación de composición no estuvo definida en

las primeras versiones de UML y por ello es que las técnicas más antiguas no la consideran. La primera columna de la Tabla 2-18 muestra el código que se le ha asignado a la técnica; la segunda columna, los autores; la tercera columna, la propuesta de la técnica para la relación de composición entre las clases E y F, las cuales tienen una relación de agregación (ver Figura 2-4) y la última columna indica si la técnica cumple o no con los últimos manuales de PF. En esta última columna, se indica con un guión (“-”) si es que la técnica no define reglas para composición.

Tabla 2-19. Propuestas de las técnicas para composición

Código	Autores	Reglas para Composición	¿Cumple con Ifpug CPM 4.2.1 y 4.3.1?
Ta1	Fetcke T., et. al	No definido	-
Ta2	Antoniol G., et. al	No definido	-
Ta3	Uemura T., et. al	No definido	-
Ta4	Cantone G., et. al	No definido	-
Ta5	Harput V., et. al	No definido	-
Ta6	Zivkovic A., et. al	Emplea propuesta de Antoniol (Ta2): Dependiendo del punto de vista, C y D pueden ser ficheros separados o uno solo por los dos. Añadir un RET a C, sin importar que multiplicidad tenga.	No, porque la relación de composición indica dependencia entre E y F, por lo que debería ser un solo fichero.
Ta7	Abrahao S., et. al	C y D son considerados como un fichero con 2 RET. Añadir DET y RET según criterio de asociación de Antoniol (Ta2)	Cumple con el criterio que sea un solo fichero. El conteo de DET y RET no cumplen.
Ta8	Lavazza L.A., et. al	Si E y F están en un componente, se cuenta un fichero con un RET	No, porque dependen si está en un solo componente y no por la relación de composición en sí.
Ta9	Chamundeswari A., et. al	Según la multiplicidad de la relación, se sigue regla de asociación de Antoniol (Ta2): Si C tiene multiplicidad de 1, a D se le añade un DET. Si es mayor a 1, a D se le añade un RET.	No, porque debería considerarse como un solo fichero.

Según lo analizado en las técnicas, solo la técnica Ta7 propuesta por Abrahao et. al [Abrahao, 2006] considera identificar un fichero para E y F; sin embargo, la determinación de DET y RET no cumple con lo establecido por la técnica de Puntos de función. Las técnicas restantes o no consideran la relación de composición o su propuesta no está de acuerdo a lo definido en los últimos manuales de Puntos de función.

2.6.1.3 Relaciones de herencia entre clases

La técnica de Puntos de función define una regla para determinar la cantidad de ficheros y RET para la relación tipo/subtipo en un diagrama E-R. Este tipo de relación es equivalente a la herencia entre clases. Por ejemplo,

pueden haber dos tipos de empleados: empleado por contrato y empleado permanente; sin embargo, ambos son empleados y comparten la misma información como nombres y apellidos, pero también tienen información diferente que no comparten. La Figura 2-5 muestra la representación del ejemplo de empleado en diagrama E-R y en diagrama de clases (el ejemplo de diagrama E-R fue tomado de [Ifpug, 2009]).

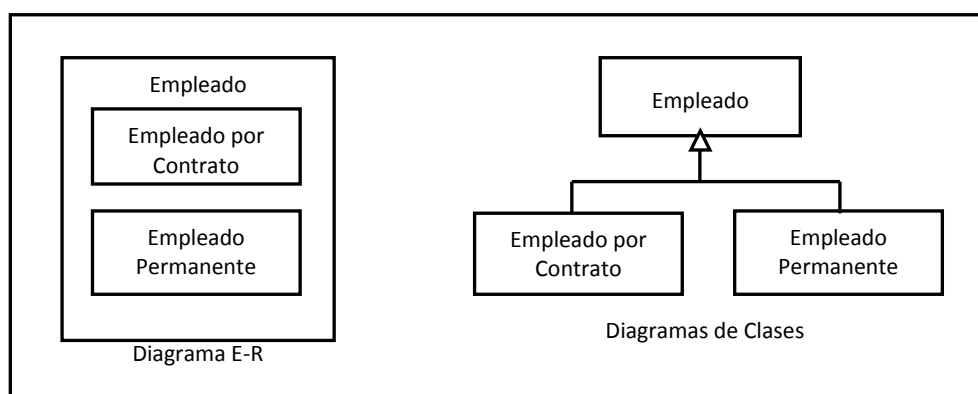


Figura 2-5. Ejemplo de subtipo y de herencia para empleado

La técnica de PF define que se debe contar un solo fichero para aquellas entidades que contienen subtipos. Además, si los subtipos tienen atributos que los diferencian, se debería contar un RET por cada subtipo para el fichero. En el caso que solo exista un atributo único en uno de los subtipos, entonces solo se cuenta un RET. Por ejemplo, si la entidad empleado solo tuviera los subtipos Empleado Casado y Empleado Soltero, y solo existiera el único atributo “nombre de esposo(a)” el cual diferenciaría a ambos subtipos, entonces se contaría un RET y no dos.

Considerando que “J” es una clase padre que tiene como hijas a “K” y a “L”, se identificó lo que propone cada técnica de PF para la herencia. La primera columna de la Tabla 2-20 muestra el código que se le ha asignado a la técnica; la segunda columna, los autores; la tercera columna, la propuesta de la técnica para la relación de herencia entre las clases J, K y L y la última columna indica si la técnica cumple o no con los últimos manuales de PF.

Tabla 2-20. Propuestas de las técnicas para herencia

Código	Autores	Generalización	¿Cumple con Ifpug CPM 4.2.1 y 4.3.1?
Ta1	Fetcke T., et. al	Objetos abstractos se cuenta como RET. Si J es abstracto y K y L son ficheros, estos últimos tienen un RET adicional por J. Si J, K y L son concretos pero solo K y L (hijos) son ficheros, adicionar un RET a estos últimos. Si J (padre) es fichero y no son ficheros K y L, adicionar a J un RET por cada hijo.	No, porque debería ser un solo fichero
Ta2	Antoniol G., et. al	Puede considerarse J junto con K como un fichero con 2 RET y J con L como un fichero con 2 RET. También podría considerarse las tres cada una como un fichero separado con un RET c/u. Depende del punto de vista del usuario o del diseñador	No, porque debería ser un solo fichero
Ta3	Uemura T., et. al	En un fichero, contar los DET de la clase que hereda	No indica claramente si es un solo fichero o varios. Solo indica la cantidad de DET que en parte sí es correcto.
Ta4	Cantone G., et. al	Usa propuestas de Fetcke (Ta1) y Antoniol (Ta2). Añade regla para herencia múltiple: 1 RET para cada super-clase	No, porque debería ser un solo fichero. La herencia múltiple no es incluida en Ifpug
Ta5	Harput V., et. al	Contar un fichero por todas las clases. Se puede considerar un RET por cada hija o, alternativamente, un RET por cada clase de la herencia.	Parcialmente, porque no se debería contar un RET por cada clase de la herencia.
Ta6	Zivkovic A., et. al	Emplea propuesta de Antoniol (Ta2)	No, porque debería ser un solo fichero
Ta7	Abrahao S., et. al	Contar un fichero por cada clase J, K y L. Añadir los DET de J para K y L.	No, porque debería ser un solo fichero
Ta8	Lavazza L.A., et. al	Si J, K y L están en un componente, solo se cuentan como RET las clases que se pueden instanciar.	No, porque debería ser un solo fichero
Ta9	Chamundeswari A., et. al	Son ficheros independientes. Asignar a cada clase un RET adicional. En las clases derivadas añadir los DET de la base. Contar un RET a las hijas por cada dato complejo que tenga el padre.	No, porque debería ser un solo fichero

Como se puede observar en la Tabla 2-20, la mayoría de las técnicas consideran determinar ficheros separados por cada clase de una herencia, solo la técnica Ta5 de Harput [Harput, 2005] indica que se debe contar como un solo fichero; no obstante, esta técnica no cuenta adecuadamente la cantidad de RET. La técnica Ta4 [Cantone, 2004] considera el caso de herencia múltiple que no es definido en PF. Esto quiere decir que ninguna de las técnicas propone reglas que cumplan con lo establecido por los últimos manuales de PF.

2.6.1.4 Clase asociación

Una clase asociación es aquella que se encuentra relacionada a una asociación entre dos clases. La clase asociación se crea cuando la asociación tiene más información como atributos adicionales [Eriksson, 2004]. En la Figura 2-6, Asignación de Trabajo es una clase asociación que pertenece a la relación de asociación de las clases Empleado y Trabajo.

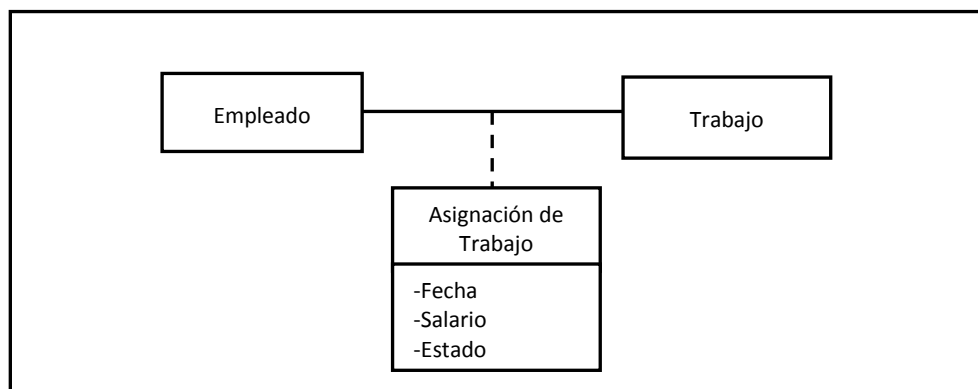


Figura 2-6. Ejemplo de clase asociación

El elemento en diagramas de clase que podría considerarse como equivalente a la “entidad asociativa” de un diagrama E-R es la “clase asociación”. A diferencia de la clase asociación; en un diagrama E-R, se puede incluir una entidad asociativa que no tenga atributos adicionales que no sea las llaves foráneas de las entidades que asocia [Ifpug, 2009]. Según lo definido en la técnica de PF, las reglas para contar entidades asociativas son las siguientes:

- a) Si la entidad asociativa no tiene atributos adicionales a las llaves foráneas de las entidades que asocia, ésta no es considerada para el cálculo de PF.
- b) Si la entidad asociativa tiene atributos adicionales, pueden ocurrir dos casos: que las reglas de negocio indiquen que la entidad asociativa debe retener su información de manera independiente a las entidades que asocia o, en caso contrario, sea dependiente. En el primer caso, se reconocería a la entidad asociativa como un fichero independiente y en el segundo caso, como RET de las entidades que asocia.

Por ejemplo, para las clases que muestra la Figura 2-6, se debería aplicar la regla definida en b). Si la Asignación de Trabajo, según las reglas de negocio, tuviera que mantenerse de manera independiente, se contarían tres ficheros y cada uno con un RET (es un fichero por cada clase). En el caso que se tuviera que mantener dependiente, entonces las clases Empleado y Trabajo contendrían un RET adicional y Asignación de Trabajo no generaría ficheros adicionales.

En el caso de diagramas de clases, la regla a) no aplicaría, ya que, a diferencia de los diagramas E-R, la clase asociación solo debe existir si es que la asociación tiene atributos adicionales.

Ninguna de las técnicas incluye en sus reglas el concepto de clase asociación. Aunque la técnica Ta5 propuesta por Harput et. al [Harput, 2005]

tampoco incluye en sus reglas la clase asociación, sí indica que una relación muchos a muchos debe ser considerada como un fichero, pero no define si esa relación debería tener atributos adicionales. En el caso que solo sea una asociación con multiplicidad muchos a muchos, debería cumplirse la regla a) que indica que la relación no tiene que ser considerada como fichero ni como RET; por lo tanto la técnica Ta4 de Harput et. al no cumpliría lo especificado por los últimos manuales de puntos de función.

2.6.1.5 Resumen de la revisión de ficheros

Todas las técnicas proponen emplear los diagramas de clases para poder identificar los ficheros de Puntos de función. Las técnicas que emplean el diagrama de clases propuesto por Jacobson [Jacobson, 1992], solo deben emplear las clases de tipo entidad.

En cuanto a los atributos, todas proponen utilizarlas para identificar la cantidad de DET de cada fichero. Sin embargo, dos técnicas [Cantone, 2004] [Chamundeswari, 2008] indican que debería contarse un RET por cada atributo complejo de cada clase. Esto último no estaría de acuerdo a los últimos manuales de Puntos de función, ya que sería equivalente a una relación en un diagrama E-R y estos podrían ser considerados como DET, pero nunca como RET.

Siguiendo las recomendaciones de Webster y Watson [Webster, 2002] para sintetizar la comparación entre las técnicas en cuanto a las reglas para determinar los ficheros en PF, se realizó la Tabla 2-21. La tabla presenta el grado de cumplimiento que tienen las diferentes propuestas de los autores a las reglas especificadas en los manuales de PF 4.2.1 y 4.3.1.

La primera columna de la tabla muestra el código de la técnica, la segunda columna presenta los autores de la técnica, la tercera columna indica qué manual de PF es empleado para definir su propuesta y las columnas siguientes muestran si proponen reglas que estén de acuerdo con los últimos manuales de PF para los siguientes elementos de un diagrama de clases: asociación, agregación composición, generalización y clase asociación. Se ha indicado con una equis ("X"), si la técnica define reglas que cumplan con los últimos manuales de PF para una relación de UML; se deja la celda en blanco si no cumple; y "Parcialmente", si hay algunos elementos que cumple, pero no todos.

Tabla 2-21. Tabla comparativa con el grado de cumplimiento de las técnicas de PF para determinar los ficheros

Código	Autores	Versión de Manual	¿Cumple con Ifpug CPM 4.2.1 y 4.3.1?				
			Asociación	Agregación	Composición	Generalización	Clase asociación
Ta1	Fetcke T., et. al	4.0					
Ta2	Antoniol G., et. al	4.0					
Ta3	Uemura T., et. al	4.1					
Ta4	Cantone G., et. al	4.1.1					
Ta5	Harput V., et. al	4.1.1				Parcialmente	
Ta6	Zivkovic A., et. al	4.1					
Ta7	Abrahao S., et. al	4.1			Parcialmente		
Ta8	Lavazza L.A., et. al	4.2					
Ta9	Chamundeswari A., et. al	4.1					

Como se puede observar en la Tabla 2-21, en relación a las diferentes relaciones que pueden existir entre clases, ninguna de las reglas propuestas por las técnicas cumplen las indicaciones de PF; por ello, no se ha incluido la equis en ninguna celda de la tabla. Sin embargo, solo las técnicas Ta6 de Harput et. al [Harput, 2005] cumple parcialmente en generalización y Ta7 de Abrahao et. al [Abrahao, 2006] cumple parcialmente en composición.

Cabe resaltar que la relación de composición no estaba definida en las primeras versiones de UML, por ello las técnicas Ta1, Ta2 y Ta3 no las consideran en sus propuestas. Sin embargo, las técnicas Ta4 y Ta5 no han incluido este tipo de relación, cuando a la fecha de publicación de estas propuestas ya UML definía lo que era composición.

En cuanto a la clase asociación, ninguna de las técnicas define reglas para este concepto. Sin embargo, solo la técnica Ta5 propuesta por Harput et. al [Harput, 2005] indica que una relación muchos a muchos debe ser considerada como un fichero, pero no define si esa relación debería tener atributos adicionales; por lo tanto no se puede considerar como clase asociación.

En resumen, se puede observar que las técnicas encontradas no cumplen con las reglas propuestas por los últimos manuales de Puntos de función [Ifpug, 2004][Ifpug, 2009]. Esto puede deberse a que todas las técnicas, a excepción de una de ellas, emplea como referencias manuales más antiguos de Puntos de función [Ifpug, 1999].

2.6.2 Transacciones de Puntos de función

Para poder determinar la cantidad de PFSA por cada transacción, sea EI, EO o EQ; primero se tiene que identificar la cantidad de DET y FTR que tiene cada transacción. Los FTR corresponden a cuántos ficheros son empleados en una transacción, sin importar si son ILF o EIF. La Tabla 2-22 muestra una síntesis de

las reglas para determinar los DET de cada transacción que especifica los manuales de PF. Se indica con una equis ("X") si la regla aplica para EI, EO o EQ.

Tabla 2-22. Reglas para determinar DET en las transacciones de PF

Nº	Regla	EI	EO	EQ
1	Contar un DET por cada campo reconocido por el usuario que ingresa o sale de los límites de la aplicación y es requerido para completar la transacción	X	X	X
2	Contar un DET para la habilidad de especificar una acción a ser tomada aún si hay múltiples métodos para invocar el mismo proceso lógico	X	X	X
3	Contar un DET para la capacidad de enviar un mensaje de respuesta fuera de los límites de la aplicación para indicar un error ocurrido durante el procesamiento, confirmar que el procesamiento está completo o verificar que el procesamiento debería continuar.	X	X	X
4	No contar campos que son obtenidos o derivados por el sistema y guardados en un ILF durante el proceso elemental si los campos no cruzan los límites de la aplicación	X	X	
5	Si un DET ingresa y sale de los límites contar sólo uno.		X	X
6	No contar literales como DET		X	X
7	No contar variables de paginación o datos generados por el sistema.		X	X

Las revisiones y comparaciones entre lo que propone cada técnica para las transacciones se realizará teniendo en cuenta los siguientes criterios: diagramas de UML empleados y metodología de desarrollo de software asociada (ver acápite 2.6.2.1); identificación de los tipos de transacciones -EI, EO, EQ- (ver acápite 2.6.2.2); y reglas adicionales para identificar DET: habilidad para especificar una acción y capacidad para envío de mensajes (ver acápite 2.6.2.3). Como parte final de esta subsección, el acápite 2.6.2.4 muestra un resumen de la revisión de las técnicas con relación a las transacciones.

2.6.2.1 Diagramas de UML empleados y metodologías de desarrollo de software

A diferencia de las propuestas para identificar ficheros en las que la mayoría emplean diagramas de clases, los autores utilizan diferentes diagramas de UML para realizar el cálculo de PF para transacciones.

La Tabla 2-23 muestra un resumen de los técnicas para identificar transacciones de PF. La primera columna de la tabla indica el código de la técnica; la segunda columna, los autores; la tercera columna, los diagramas de UML empleados; la cuarta columna, las reglas definidas para identificar DET; y, finalmente, la quinta columna, las reglas para identificar FTR.

Tabla 2-23. Propuestas de la técnicas para identificar y contar transacciones

Código	Autores	Diagramas de UML Empleados	DET	FTR
Ta1	Fetcke T., et. al	Ninguno	Emplear DET de ficheros	Ficheros mantenidos y/o leídos en un caso de uso
Ta2	Antoniol G., et. al	Emplea métodos concretos de las clases, mas no los abstractos.	Datos simples en argumentos de métodos y variables globales simples	Argumentos complejos, objetos o variables globales complejas referenciadas por los métodos
Ta3	Uemura T., et. al	Diagramas de secuencia. Utiliza solo los ficheros como objetos del diagrama	Argumentos de los mensajes	Ficheros que participan en el diagrama
Ta4	Cantone G., et. al	Diagramas de secuencia con estereotipos OOSE de Jacobson [Jacobson, 1992] y RUP [IBM, 2006]	Argumentos de los mensajes que van hacia los objetos entidad	Objetos entidad del diagrama de secuencias
Ta5	Harput V., et. al	Diagramas de secuencia	Argumentos de los mensajes de un diagrama de secuencias	Clases que son ficheros y que son argumentos en diagramas de secuencias. Clases que son referenciadas o mantenidas en las descripción de un requerimiento funcional
Ta6	Zivkovic A., et. al	Diagramas de actividades que describe un caso de uso. Las actividades definidas en el <i>swimlane</i> del sistema son consideradas transacciones	Igual a propuesta de Ta2 de Antoniol	Igual a propuesta de T2 de Antoniol
Ta7	Abrahao S., et. al	Diagramas de transición de estados y diagramas de comunicación.	DET es argumento del servicio, atributo en un patrón de presentación	Clase empleada por un argumento del servicio o clase empleada en diagrama de interacción de objetos, patrones de presentación o diagrama de transición de estados
Ta8	Lavazza L.A., et. al	Diagramas de secuencia en el que se incluye como objetos al Sistema y a los ficheros	Argumentos de los mensajes que cruzan los límites de la aplicación	Ficheros que son referenciados en el diagrama de secuencias
Ta9	Chamunde swari A., et. al	Emplea métodos de las clases	Valor de retorno o argumento de un mensaje de tipo simple (int, float, etc.) o void	Valor de retorno o argumento de un mensaje de tipo complejo (objeto, eventos, etc.)

Como se puede observar en la Tabla 2-23, las técnicas difieren en especificar qué tipo de diagrama de UML se debería utilizar para identificar transacciones. Cuatro de las técnicas emplean diagramas de secuencia; una de ellas, diagramas de comunicación; una, diagramas de estado; y una, diagramas de actividades. También, se puede observar que dos técnicas solo emplean los métodos definidos en las clases y no otro diagrama adicional. La técnica Ta1 de Fetcke et. al, no emplea ningún tipo de diagrama, pero sí considera el contenido de la especificación de casos de uso. Adicionalmente se puede observar que las técnicas Ta3 de Uemura et. al y Ta8 de Lavazza et. al usan diagramas de secuencia que solo serían útiles para el cálculo de PF, porque añaden los ficheros PF como objetos del diagrama, y no para apoyar el diseño del sistema; tal y como lo proponen metodologías como RUP [IBM, 2006] o el método de Jaaksi [Jaaksi, 1998].

En cuanto a la identificación de DET y FTR de las transacciones, sin considerar a la técnica Ta1 de Fetcke et. al, todas tienen elementos comunes para identificar los DET y FTR. En cuanto a la determinación de DET, las técnicas los identifican a partir de los argumentos de los mensajes o de los métodos de los objetos. Sobre los FTR, las técnicas consideran como tal a aquellas clases u objetos que participan en el envío de mensajes en una transacción; algunas de ellas indican explícitamente que solo son aquellas clases u objetos que son reconocidos como ficheros.

Al revisar las técnicas, se pudo observar que el empleo de diagramas para identificar las transacciones era dependiente de la metodología de desarrollo que podrían estar empleando o sería una adaptación de algún diagrama que propondría alguna metodología. La Tabla 2-24 muestra los diagramas de UML empleados y a qué metodología podrían estar asociadas. La primera columna de la tabla indica el código de la técnica; la segunda columna, los autores; la tercera columna, los diagramas de UML empleados para contar los PF correspondientes a las transacciones; y la última columna muestra la metodología o método que podría estar asociado el diagrama UML empleado. En esta última columna, se ha colocado un guión ("-") si la técnica no utiliza ningún diagrama de UML para este fin.

Tabla 2-24. Diagramas empleados y metodología relacionada

Código	Autores	Diagramas de UML Empleados	Metodología asociada
Ta1	Fetcke T., et. al	Ninguno	-
Ta2	Antoniol G., et. al	Emplea métodos concretos de las clases, mas no los abstractos.	-
Ta3	Uemura T., et. al	Diagramas de secuencia. Utiliza solo los ficheros como objetos del diagrama	Parecido a diagramas de secuencias de Larman [Larman, 1999] pero añade ficheros
Ta4	Cantone G., et. al	Diagramas de secuencia con estereotipos OOSE de Jacobson [Jacobson, 1992] y RUP [IBM, 2006]	OOSE de Jacobson [Jacobson, 1992] y RUP [IBM, 2006]
Ta5	Harput V., et. al	Diagramas de secuencia	No especifica metodología
Ta6	Zivkovic A., et. al	Diagramas de actividades que describe un caso de uso. Las actividades definidas en el <i>swimlane</i> del sistema son consideradas transacciones	Diagramas de actividades es opcional en muchas metodologías [IBM, 2006]
Ta7	Abrahao S., et. al	Diagramas de transición de estados y diagramas de comunicación.	OO-method [Pastor, 2001]
Ta8	Lavazza L.A., et. al	Diagramas de secuencia en el que se incluye el Sistema como un objeto y a los ficheros	Parecido a diagramas de secuencias de Larman [Larman, 1999] pero añade ficheros
Ta9	Chamundeswari A., et. al	Emplea métodos de las clases	-

De lo que se puede observar de la Tabla 2-24, hay tres técnicas que no emplean diagramas de UML: Ta1, Ta2 y Ta9. Las técnicas Ta3 de Uemura et. al y Ta8 de Lavazza et. al se basan en la metodología propuesta por Larman [Larman, 1999] que indica que en paralelo al diagrama de clases se debería hacer un diagrama de secuencias del sistema, pero le añade a este diagrama los ficheros de puntos de función que se han identificado. Cantone et. al que propone la técnica Ta4 utilizan los diagramas de secuencias propuestos por OOSE de Jacobson [Jacobson, 1992] y RUP [IBM, 2006], en el que se deben incluir los estereotipos de clases *boundary*, *control* y *entity*. La propuesta de Harput et. al indica que se deben utilizar diagramas de secuencia, pero no se puede determinar a qué metodología se refiere; sin embargo, se podría relacionar a metodologías como Larman, OOSE o RUP que sí proponen la utilización este tipo de diagramas. Abrahao et. al (técnica Ta7) considera el empleo de una metodología propia del grupo de investigación donde pertenecen que se denomina OO-method. Finalmente, la técnica Ta6 de Zivkovic et. al propone emplear diagramas de actividades, aunque este tipo de diagramas es considerado opcional por muchas metodologías

El diagrama de secuencias es el diagrama más utilizado por las técnicas para hacer el cálculo de PF debido a las transacciones. Sin embargo, no todas las metodologías de desarrollo orientadas a objetos emplean este tipo de diagramas en las etapas iniciales de un proyecto; por ejemplo Jaaksi [Jaaksi, 1998] emplea diagramas de secuencia solo en el diseño. Abrahao et. al, considera el empleo de diagramas de comunicación, el cual cumple el mismo objetivo que los diagramas de secuencia. Los estereotipos de clases *boundary*, *control* y *entity* que utiliza Cantone et. al en Ta4 son empleados por algunas metodologías como OOSE [Jacobson, 1992] y RUP [IBM, 2006]; sin embargo existen muchas metodologías que solo utilizan las clases consideradas como de tipo *entity*.

Como se puede observar, hay mucha variación en el empleo de diagramas de UML para cálculo de transacciones de PF debido a qué metodología base están considerando los autores. Por ello, sería recomendable realizar una propuesta que se ajuste a la mayoría de metodologías de software existentes.

2.6.2.2 Identificación de tipo de transacciones

Luego de revisar qué diagramas de UML emplean las técnicas para determinar las transacciones de Puntos de función y qué metodología base podrían estar considerando, se evaluó cuáles son las reglas que emplean las técnicas para identificar el tipo de transacción (EI, EO o EQ). La síntesis de lo que propone cada técnica en este aspecto se incluye en la Tabla 2-25. En la primera columna de la tabla se indica el código de la técnica; en la segunda columna, los autores; y en la columna final, las reglas para identificar el tipo de transacción de Puntos de función.

Tabla 2-25. Propuestas para identificar tipos de transacciones

Código	Autores	Identificación de EI, EO, EQ
Ta1	Fetcke T., et. al	Seguir reglas de Ifpug
Ta2	Antoniol G., et. al	No los diferencia
Ta3	Uemura T., et. al	Identifica tipo de transacción según los mensajes del diagrama de secuencias. Define patrones de diagramas
Ta4	Cantone G., et. al	Se tiene que identificar que secuencias de mensajes pueden ser transacciones. Luego, dependiendo los patrones definidos por la técnica, se puede determinar que tipo de transacción es.
Ta5	Harput V., et. al	Seguir reglas de Ifpug
Ta6	Zivkovic A., et. al	No los diferencia. Usa una sola tabla de complejidad para todas las transacciones
Ta7	Abrahao S., et. al	Sí, según elementos de OO-Method: EI es un servicio de una clase o <i>legacy view</i> , EO y EQ son instancias según patrones de interacción.
Ta8	Lavazza L.A., et. al	Se indica en el diagrama de secuencias que mensaje es el propósito principal del sistema, mediante una nota
Ta9	Chamundeswari A., et. al	No los diferencia

Según lo indicado en la Tabla 2-25, se puede observar que tres técnicas (Ta3, Ta6 y Ta9) no incluyen reglas para determinar el tipo de transacción y considera a EI, EO y EQ como si fuera un mismo tipo. Esto va en contra de lo especificado por la técnica de PF, porque cada tipo de transacción tiene un peso diferente en cuanto a cantidad de PFSA.

Las técnicas Ta3 de Uemura et. al y Ta4 de Cantone et. al consideran que se puede identificar el tipo de transacción comparando patrones de interacción en diagramas de secuencias, patrones que ambas técnicas definen como guías para definir si es EI, EO o EQ. La técnica Ta8 de Lavazza et. al también emplea diagramas de secuencia, pero no define patrones de interacción como Ta3 y Ta4.

En el caso de la técnica Ta7 de Abrahao et. al, se reconoce los tipos de transacciones según los elementos que especifica la metodología OO-method y que se han utilizado para especificar el software que se va a desarrollar. Dos técnicas, Ta1 de Fetcke et. al y Ta5 de Harput et. al, indican que se deben seguir las reglas que tiene la técnica de PF para identificar el tipo de transacciones.

En resumen, la mayoría de las técnicas tienen reglas que identifican apropiadamente el tipo de transacciones. Solo tres de ellas no toman en cuenta las diferencias y solo identifican si son transacciones o no.

2.6.2.3 Reglas adicionales para identificar DET

Existen dos reglas relacionadas a DET de transacciones que se incluyen a partir del manual del año 1999 [Ifpug, 1999] y que se indican en la Tabla 2-22. Estas dos reglas las denominaremos como regla 2 y regla 3. La regla 2 especifica que se debe contar un DET para la habilidad de especificar una acción a ser tomada y la regla 3, que se debe contar un DET para la capacidad de enviar un mensaje de respuesta fuera de los límites de la aplicación para indicar un error

ocurrido durante el procesamiento, confirmar que el procesamiento está completo o verificar que el procesamiento debería continuar. La Tabla 2-26 muestra el resultado de la revisión de las técnicas en relación a estas dos reglas. La primera columna de la tabla indica el código de la técnica; la segunda, los autores; la tercera, si cumple o no la regla 2; y la última columna, si cumple la regla 3.

Tabla 2-26. Propuestas para identificar tipos de transacciones y reglas 2 y 3 para identificar DET

Código	Autores	¿Cumple regla 2 para DET?	¿Cumple regla 3 para DET?
Ta1	Fetcke T., et. al	No	No
Ta2	Antoniol G., et. al	No	No
Ta3	Uemura T., et. al	No	No
Ta4	Cantone G., et. al	No	No
Ta5	Harput V., et. al	No	Sí, mensaje del diagrama de secuencias
Ta6	Zivkovic A., et. al	No	No
Ta7	Abrahao S., et. al	Sí	Sí
Ta8	Lavazza L.A., et. al	Sí	No se indica claramente
Ta9	Chamundeswari A., et. al	No	No

En cuanto al cumplimiento de las reglas 2 y 3 referentes a DET, solo la técnica Ta7 de Abrahao et. da detalles para cumplir ambas reglas, la técnica Ta5 de Harput et. al solo cumple la regla 3 y la técnica Ta8 cumple la regla 2. Esta última técnica no indica claramente cómo cumplir con la regla 3; sin embargo, sí menciona algo que podría considerarse como parte de esta regla: debe contarse el mensaje de retorno del diagrama de secuencias como un DET. De acuerdo a lo que se puede observar, la mayoría de las técnicas no cumplen con lo que indica la regla 2 y la regla 3.

2.6.2.4 Resumen de la revisión de transacciones

Siguiendo como base las recomendaciones de Webster y Watson [Webster, 2002] para sintetizar y comparar lo que proponen diferentes técnicas, se realizó la Tabla 2-27. La tabla muestra la comparación de estas técnicas en base a los siguientes criterios: identificación de tipo de transacción según lo especificado en los manuales de Puntos de función (ver tercera columna), no dependencia de una metodología de desarrollo de software orientado a objetos específica (ver cuarta columna), no propone diagrama que solo sea empleado para el cálculo de PF (ver quinta columna), identificación de DET según los últimos manuales de Puntos de función (ver séptima, octava y novena columna) y, finalmente, identificación de FTR de acuerdo a los últimos manuales de Puntos de función.

En la Tabla 2-27, se ha indicado con una equis ("X"), si la técnica cumple con el criterio definido que se indica en la cabecera de la tabla; se deja la celda en

blanco si no cumple; y “Parcialmente”, si hay algunos elementos que cumple, pero no todos.

Tabla 2-27. Tabla comparativa de lo que proponen las técnicas para determinar las transacciones de PF

Cód.	Autores	Identificación de EI, EO, EQ según Ifpug CPM	No crea diagrama solo para cálculo de PF	No depende de metodología de desarrollo	Identificación de DET de acuerdo a Ifpug CPM			Identificación de FTR de acuerdo a Ifpug CPM
					DET información	Cap. Mensaje	Habilidad Acción	
Ta1	Fetcke T., et. al	X	X	X	X			X
Ta2	Antoniol G., et. al		X	X	X			Parcialmente
Ta3	Uemura T., et. al	X			X			X
Ta4	Cantone G., et. al	X	X		X			X
Ta5	Harput V., et. al	X	X		X	X		X
Ta6	Zivkovic A., et. al		X		X			Parcialmente
Ta7	Abrahao S., et. al	X	X		X	X	X	X
Ta8	Lavazza L.A., et. al	X			X	X	X	X
Ta9	Chamundeswari A., et. al		X	X	X			Parcialmente

Como se puede observar en la Tabla 2-27, todas las técnicas no cumplen con los criterios de comparación definidos. Sin embargo, es importante que existe una técnica que cumple todos los criterios, a excepción de la no dependencia de metodología. Esta técnica es la Ta7 propuesta por Abrahao et. al.

2.6.3 Conclusiones de la revisión de técnicas para calcular puntos de función en modelos orientados a objetos

Se han propuesto varias técnicas para hacer frente al cálculo de PF en desarrollos de software orientado a objetos. Sin embargo, muchas de ellas no consideran lo especificado en los manuales de PF definidas por el Ifpug. Algunas de aquellas técnicas que sí consideran las reglas especificadas por el Ifpug, han sido diseñadas antes de que salieran los últimos manuales [Fetcke, 1997] [Antoniol, 1999] [Uemura, 2001] [Cantone, 2004] [Abrahao, 2006]. Existen técnicas que han sido definidas después de que se publicaran los últimos manuales de PF; pero a pesar de ello, consideran lo especificado en manuales antiguos [Harput, 2005] [Zivkovic, 2005] [Chamundeswari, 2008]. Solo una de ellas toma como base uno de los últimos manuales de PF [Lavazza, 2008].

Todas las técnicas utilizan diagramas de clases para realizar el cálculo de ficheros de PF. Sin embargo, la mayoría de ellas no cumplen con lo especificado

en los últimos manuales de PF y las relaciones entre clases definidas en UML [OMG, 2008].

Debido a la poca variación en las reglas para calcular las transacciones de PF, la mayoría de las técnicas cumple con casi todo lo especificado en los manuales publicados por el Ifpug. Pero, no existe consenso en qué diagrama de UML se debe utilizar para realizar este tipo de cálculo y es muy dependiente de la metodología de desarrollo orientado a objetos seguida.

Por lo tanto, debería proponerse una nueva técnica que permita realizar el cálculo de PF para desarrollos orientados a objetos que cumpla con lo siguiente: que esté de acuerdo con las reglas definidas en los últimos manuales de PF [Ifpug, 2004] [Ifpug, 2009], que considere todas las relaciones entre clases definidas por UML [OMG, 2008] y que sea, en lo posible, independiente de los diagramas propuestos por las metodologías de desarrollo de software, de tal manera que pueda ser aplicable en cualquiera de ellas. También, en el caso de las transacciones, se debería considerar en no añadir diagramas que sólo puedan ser utilizados para el cálculo de PF, pero no se puedan reutilizar en etapas posteriores como el diseño.

2.7 Técnicas para Calcular el Esfuerzo Requerido por cada Incremento

Tras revisar los artículos seleccionados del proceso de búsqueda, se le asignó un código a cada técnica para que sea más fácil su identificación durante toda la revisión. Por ejemplo, la primera técnica tiene el código “Tb1”; porque la letra “T” significa que es una **Técnica**, la letra “b” viene de la codificación de la pregunta de investigación relacionada que es **2b** y el número final es un correlativo. La primera columna de la Tabla 2-15 muestra el código asignado a cada técnica; la segunda columna, los autores; la tercera columna, el título del artículo; la cuarta columna, el tipo de artículo; y la última columna, el año en que se publicó el artículo.

Tabla 2-28. Técnicas identificadas para cálculo de esfuerzo requerido para cada incremento

Código	Autores	Título	Tipo de publicación	Año
Tb1	Orr G., Reeves T.E.	Function point counting: One program's experience [Orr, 2000]	Journal	2000
Tb2	Kang S., Choi O., Baik J.	Model-based dynamic cost estimation and tracking method for agile software development [Kang, 2010]	Conferencia	2010
Tb3	Hericko M., Zivkovic A.	The size and effort estimates in iterative development [Hericko, 2007]	Journal	2007

Las técnicas se evaluaron empleando los siguientes criterios: definición de priorización para la construcción de casos de uso (ver subsección 2.7.1), métricas basadas en PF empleada para estimar el tamaño de cada incremento (ver subsección 2.7.1), modelos de conversión utilizados para calcular puntos de función en orientación a objetos si es que utilizara (ver subsección 2.7.2) y cómo se realiza la estimación del esfuerzo de cada incremento (ver subsección 2.7.3). La subsección 2.7.4 muestra las conclusiones de esta revisión.

2.7.1 Priorización de casos de uso y métricas para estimar el tamaño de software

Ninguna de las tres propuestas indica cómo se deben priorizar los casos de uso o requisitos de software de tal manera que se puedan determinar qué se construye en cada incremento. Por ello, se tuvo que realizar otra búsqueda para encontrar técnicas de priorización de casos de uso. La siguiente sección, Sección 2.8, muestra la revisión de las técnicas encontradas para este propósito.

En cuanto a las métricas empleadas para estimar el tamaño de cada incremento, las técnicas Tb2 de Kang et. al [Kang, 2010] y Tb3 de Hericko [Hericko, 2007] utilizan los PFSA. En cambio, la técnica Tb1 de Orr et. al [Orr, 2000] hace una comparación de la eficiencia de la técnica original de Puntos de función a nivel de análisis, versus la eficiencia de propuestas de Puntos de función para diseño e implementación.

Existen críticas sobre si se debería emplear el factor de ajuste en la técnica de PF, algunas de ellas son indicadas por Lokan [Lokan, 2000]. También, hay estudios que no emplean este factor de ajuste y utilizan solo los PFSA con buenos resultados [Koh, 2008]. Jeffery et al. [Jeffery, 1993] muestra que el no utilizar el factor de ajuste no afecta en la precisión de la estimación del esfuerzo. Cocomo [Boehm, 2000a], uno de los modelos de estimación de costos mejor documentados y conocidos, permite emplear los PFSA para estimar el esfuerzo de un proyecto de desarrollo de software.

2.7.2 Modelo de conversión de Puntos de función a orientación a objetos

Solo una de las técnicas emplea un modelo de conversión para calcular PF con modelos orientados a objetos. Esta es la técnica Tb3 de Hericko et. al [Hericko, 2007] que utiliza como base un trabajo previo que ellos mismos realizaron [Zivkovic, 2005] y que ha sido revisada y analizada en la sección anterior (ver Sección 2.6).

2.7.3 Estimación del esfuerzo entre incrementos o iteraciones

La técnica Tb1 de Orr et. al [Orr, 2000] no explica como estimar el esfuerzo de cada iteración, en cambio las otras dos técnicas, Tb2 y Tb3, sí dan propuestas de cómo hacer esta tarea.

Ninguna de las técnicas indica cómo se deben distribuir los PF entre los diferentes incrementos. Un problema que no aborda ninguna de las propuestas es cómo repartir los PF pertenecientes a los ficheros. Por ejemplo, si hubieran dos casos de uso CU1 y CU2 que actualizan un mismo fichero ILF_A, y el CU1 es implementado en el incremento 1 y el CU2 es desarrollado en el incremento 2 ¿cómo se repartirían los PF del ILF_A entre el incremento 1 y el 2? Este tipo de interrogantes no son explicadas por ninguna técnica. El caso de las transacciones es más simple, porque se asignan cada una de ellas al incremento en el cual se implementará; pero incluso para este caso, las técnicas no indican cómo repartir los PF de transacciones entre los incrementos o iteraciones.

La técnica Tb3 de Hericko et al emplea la diferencia entre el valor estimado y el valor real del tamaño del software construido en una iteración previa para predecir el tamaño del software de la siguiente iteración. El tamaño de la siguiente iteración es la suma del tamaño estimado del software de esa iteración más un valor de corrección que se calcula en base a los estimados de tamaño de iteraciones previas. Para estimar el esfuerzo requerido de una iteración utiliza la siguiente fórmula:

$$\text{Esfuerzo} = \text{Tamaño} * \text{Productividad}$$

Los ejemplos incluidos en la publicación de la técnica Tb3 emplean el mismo ratio de productividad para todas las iteraciones: 0.7 horas/PF. El tamaño del software corresponde al tamaño estimado para esa iteración considerando la información de iteraciones previas.

La técnica Tb2 de Kang et. al [Kang, 2010] actualiza el tamaño del software en PF de cada iteración y para ello emplea el algoritmo de filtro de Kalman [Kalman, 1960]. Aunque el algoritmo empleado es diferente al de la técnica Tb3 de Hericko et. al, ambos se centran en la estimación del tamaño del software con PF.

Al revisar ambas técnicas se pudo observar que los ajustes y estimaciones se centran en el tamaño del software, pero no se toma en cuenta que la productividad podría cambiar entre iteraciones debido a diversos factores como rotación de personal, experiencia en desarrollo, etc.

2.7.4 Conclusiones de la revisión de técnicas para calcular el esfuerzo requerido por cada incremento

La Tabla 2-29 muestra un cuadro comparativo resumen de las revisiones realizadas entre las tres técnicas seleccionadas, la cual se ha realizado siguiendo las recomendaciones de Webster y Watson [Webster, 2002] para sintetizar la comparación entre propuestas o conceptos.

La primera columna de la Tabla 2-29 muestra el código de la técnica y la segunda columna, sus autores. En la tercera columna se indica si la técnica incluye alguna propuesta para priorizar la construcción de casos de uso o requerimientos entre incrementos. Las cuatro últimas columnas corresponden a

criterios de estimación del esfuerzo entre iteraciones e incrementos. En todas las columnas se ha indicado con una equis ("X"), si la técnica cumple con el criterio definido y se deja la celda en blanco si no lo cumple.

Tabla 2-29. Tabla comparativa entre técnicas para estimación del esfuerzo para cada iteración o incremento

Técnica	Autores	Priorización de Casos de Uso o Requisitos de Software	Cálculo de Esfuerzo entre Incrementos y/o Iteraciones			
			Define Reglas para Estimación	Define como dividir PF de ficheros entre incrementos	Define cómo dividir PF de transacciones entre incrementos	Considera Cambios de Productividad entre Iteraciones
Tb1	Orr G., Reeves T.E.					
Tb2	Kang S., Choi O., Baik J.		X			
Tb3	Hericko M., Zivkovic A.		X			

Como se puede observar en la Tabla 2-29, ninguna de las técnicas cumplen con todos los criterios que se han establecido. Ninguna de ellas considera cómo priorizar la construcción de casos de uso o requisitos, por ello se tuvo que hacer una búsqueda para encontrar técnicas que indiquen como realizar esta tarea (ver sección 2.8 para más detalles de técnicas de priorización).

Sobre el cálculo de incrementos, solo dos de las técnicas, Tb2 y Tb3, define reglas para realizar la estimación entre incrementos e iteraciones. Sin embargo, ninguna de las técnicas define cómo dividir los PF calculados entre iteraciones o incrementos y no considera en sus propuestas que haya cambios en la productividad del equipo de desarrollo. Esto último es importante tenerlo en cuenta en la estimación, porque la productividad entre iteraciones e incrementos podría aumentar o disminuir por diversos factores como rotación de personal, incremento en la experiencia para desarrollar software, incremento en el conocimiento de las herramientas y lenguajes con el transcurrir del tiempo, etc.

Por lo tanto, debería proponerse una nueva técnica que permita calcular el esfuerzo que se requiere para cada incremento que cumpla con lo siguiente: defina cómo priorizar los casos de uso a desarrollar, defina claramente cómo repartir los PF del software a construir entre incrementos y tome en cuenta el cambio de productividad que se podría producir mientras se avanza con la construcción de incrementos para estimar el esfuerzo.

Adicionalmente, aunque no formó parte de los criterios de comparación definidos para comparar las técnicas para calcular el esfuerzo requerido para cada incremento, se pudo observar que la técnica Tb3 de Hericko et. al es la

única que considera reglas para calcular PF tomando en cuenta modelos orientados a objetos.

2.8 Técnicas para priorizar o definir la secuencia de construcción de casos de uso

Al revisar los artículos seleccionados, se pudo observar que algunas de las técnicas tienen un nombre asignado como *Analytical Hierarchy Process (AHP)* [Perini, 2007], *B-tree* [Beg, 2008] *Multi-Person Decision-Making* [Gaur, 2010], *Planning Game* [Karlsson, 2007], *Cumulative Voting (CV)* [Berander, 2006], *Evolve* [Ruhe, 2005], *EasyWinWin* [Gruenbacher, 2000] o *Value-oriented prioritization framework (VOP)* [Mohamed, 2008]. También existen aproximaciones que mezclan técnicas anteriormente propuestas; por ejemplo, *Hierarchical Cumulative Voting (HCV)* que emplea AHP y CV [Berander, 2006]. Además, existen técnicas que utilizan requisitos no funcionales para priorizar requisitos, como la confiabilidad (*reliability* en inglés) [Runeson, 1998] o la seguridad [Peeters, 2007].

También se pudo notar que la mayoría de las técnicas, considera la opinión de los diferentes involucrados o *stakeholders* de un proyecto de software; teniendo en cuenta, principalmente, la perspectiva y punto de vista de los usuarios y clientes. Sin embargo, ninguna propone técnicas para priorizar o definir la secuencia de construcción de requisitos tomando en consideración la facilidad de construcción desde el punto de vista del desarrollador.

La construcción de software podría volverse más difícil o tomarle más tiempo al desarrollador, si es que no se toman en cuenta las precedencias o las dependencias entre requisitos. Por ejemplo, para un sistema de ventas con dos casos de uso “Registrar pedido” y “Registrar pago de un pedido”, en el cual el primero tiene una precedencia o dependencia con el segundo; si el desarrollador decidiera construir primero “Registrar pago de un pedido”, tendría que implementar algunas partes relacionadas a “Registrar pedido” porque éste aún no ha sido desarrollado. Incluso, le serían más complicadas las pruebas funcionales, porque tendría que registrar información directamente en la base de datos para poder probar el registro de pago de un pedido. Esto sería diferente si se construyera en el orden inverso.

No obstante, muy pocas consideran las precedencias o dependencias entre casos de uso, historias de usuario o requisitos de software como criterio para priorizar; o no indican explícitamente que técnica emplean para definir esta dependencia. La Tabla 2-30 muestra aquellas técnicas de las publicaciones identificadas que sí consideran precedencias o dependencias entre requisitos. Desde la primera hasta la quinta columna de la tabla, se muestra la información de los artículos que consideran dependencias entre requisitos o casos de uso; la sexta columna indica si define una técnica propia para determinar la dependencia o precedencias entre requisitos; y la última columna indica si emplea técnica de dependencias de otros autores, en el caso que no defina una propia.

Tabla 2-30. Publicaciones que consideran las dependencias o precedencias de casos de uso o requisitos

Nº	Autores	Título	Tipo de publicación	Año	Define Técnica de Precedencias o Dependencias	Emplea Técnica de Otros Autores
1	Del Sagrado J., Aguila I.M., Orellana F.J.	Requirements interaction in the next release problem	Conferencia	2011	No	No
2	Marjaie S.A., Kulkarni V.	Recognition of hidden factors in requirements prioritization using factor analysis	Conferencia	2010	No	No
3	Tonella P., Susi A., Palma F.	Using interactive GA for requirements prioritization	Conferencia	2010	No	No
4	Barney S., Hu G., Aurum A., Wohlin C.	Creating software product value in China	Journal	2009	No	No
5	Greer D., Ruhe G.	Software release planning: An evolutionary and iterative approach	Journal	2004	No	No
6	Ruhe G., Momoh J.	Strategic release planning and evaluation of operational feasibility	Conferencia	2005	No	No
7	Van Valkenhoef G., Tervonen T., De Brock B., Postmus D.	Quantitative release planning in extreme programming	Journal	2011	No	Sí
8	Barney S., Aurum A., Wohlin C.	A product management challenge: Creating software product value through requirements selection	Journal	2008	No	Sí
9	Laurent P., Cleland-Huang J., Duan C.	Towards automated requirements triage	Conferencia	2007	No	Sí
10	Berander P., Jonsson P.	Hierarchical Cumulative Voting (HCV) prioritization of requirements in hierarchies	Journal	2006	No	Sí
11	Helming J., Koegel M., Hodaie Z.	Towards automation of iteration planning	Conferencia	2009	No	Sí*
12	Benestad H.C., Hannay J.E.	A comparison of model-based and judgment-based release planning in incremental software projects	Conferencia	2011	No	Sí**
13	Some S.S.	Specifying use case sequencing constraints using description elements	Conferencia	2007	Sí	-
14	Kundu D., Samanta D.	A novel approach of prioritizing use case scenarios	Conferencia	2007	Sí	-
15	Carlshamre P., Sandahl K., Lindvall M., Regnell B., Natt och Dag J.	An industrial survey of requirements interdependencies in software product release planning	Conferencia	2001	Sí	-
16	Moisiadis Frank	Prioritizing use cases and scenarios	Journal	2000	Sí	-

Nº	Autores	Título	Tipo de publicación	Año	Define Técnica de Precedencias o Dependencias	Emplea Técnica de Otros Autores
17	Van Den Berg K.G., Simons A.J.H.	Control-flow semantics of use cases in UML	Journal	1999	Sí	-
18	Balikuddembe J.K., Potgieter A.E.	Using actor object operations structures to understand project requirements complexities	Conferencia	2008	Sí	-

* La referencia no tiene una técnica específica para determinar dependencias

** Las referencias no tienen técnicas específicas pero hacen referencia a los trabajos de Calrshamre [Carlshamre, 2002]

Como se puede observar en la Tabla 2-30, las publicaciones 1 hasta la 6, a pesar que indican que se debe tomar en cuenta las dependencias o precedencias de casos de uso, no definen una técnica para determinarlas ni emplean técnicas de otros autores para este propósito. Los artículos desde el 7 hasta el 10 consideran importante las dependencias o precedencias entre requisitos y aunque no tienen una técnica propia para determinarlas, sí emplean otras técnicas. En cuanto a la publicación 11, no tiene técnica propia, pero a la que hace referencia [Saliu, 2007] no indica como identificar las precedencias o dependencias. Algo diferente ocurre con la publicación 12, porque aunque sus referencias no tienen técnicas propias, éstas sí emplean la propuesta de Calrshamre [Carlshamre, 2002]. Las publicaciones 13 a la 18 de la Tabla 2-30 sí proponen técnicas propias.

Tras realizar la búsqueda secundaria, que corresponde a revisar las referencias y citaciones de las publicaciones 13 a la 18 de la Tabla 2-30, se obtuvieron 8 artículos más. Finalmente, la lista completa de los artículos a revisar se muestran en la Tabla 2-31. En la tabla, la primera columna muestra el código que se le ha asignado a la técnica para que se pueda referenciar con mayor facilidad (en el código, la "T" es por "técnica", "c" porque corresponde a la pregunta de investigación 2c y el número es un correlativo); la segunda columna, los autores de la técnica; la tercera columna, el título de la publicación relacionada a la técnica; la cuarta columna, si la publicación es artículo de conferencia o *journal*; la quinta columna, el año de publicación; y la última columna indica si la publicación se ha obtenido de la búsqueda primaria o secundaria. Si la publicación se ha obtenido en la búsqueda secundaria, se detalla si fue obtenida de una referencia o citación y de qué publicación.

Tabla 2-31. Técnicas para determinar dependencias o precedencias de casos de uso o requisitos

Código	Autores	Título	Tipo de publicación	Año	Fuente
Tc1	Gomez, A., Rueda, G., Alarcón, P.P.,	A Systematic and Lightweight Method to Identify Dependencies between User Stories [Gomez, 2010]	Conferencia	2010	Busqueda secundaria. Citacion de [Carlshamre, 2001]
Tc2	Sapna, P.G.; Mohanty, H.	Prioritizing Use Cases to Aid Ordering of Scenarios [Sapna, 2009]	Conferencia	2009	Busqueda secundaria. Citacion de [Kundu, 2007] y [Moisiadis, 2000]
Tc3	Balikuddembe J.K., Potgieter A.E.	Using actor object operations structures to understand project requirements complexities [Balikuddembe, 2008]	Conferencia	2008	Búsqueda primaria
Tc4	Szoke, A.	A Proposed Method for Release Planning from Use Case-based Requirements Specification [Szoke, 2008]	Conferencia	2008	Busqueda secundaria. Citacion de [Carlshamre, 2001]
Tc5	Some S.S.	Specifying use case sequencing constraints using description elements [Some, 2007]	Conferencia	2007	Búsqueda primaria
Tc6	Kundu D., Samanta D.	A novel approach of prioritizing use case scenarios [Kundu, 2007]	Conferencia	2007	Búsqueda primaria
Tc7	Whittle, J.,	Specifying Precise Use Cases with Use Case Charts [Whittle, 2007]	Conferencia	2006	Busqueda secundaria. Referencia de [Some, 2007]
Tc8	Zhang, W., Mei, H., Zhao, H.	A feature-oriented approach to modeling requirements dependencies [Zhang, 2005]	Conferencia	2005	Busqueda secundaria. Citacion de [Carlshamre, 2001]
Tc9	Denne, M., Cleland-Huang, J.	The Incremental Funding Method, A Data Driven Approach to Software Development [Denne, 2004]	Journal	2004	Busqueda secundaria. Referencia de [Laurent, 2007]
Tc10	Carlshamre, P.	Release planning in market-driven software product development: provoking an understanding [Carlshamre, 2002]	Journal	2002	Busqueda secundaria. Referencia de [Van Valkenhoef, 2011]
Tc10	Carlshamre P., Sandahl K., Lindvall M., Regnell B., Natt och Dag J.	An industrial survey of requirements interdependencies in software product release planning [Carlshamre, 2001]	Conferencia	2001	Búsqueda primaria
Tc11	Moisiadis Frank	Prioritizing use cases and scenarios [Moisiadis, 2000]	Journal	2000	Búsqueda primaria
Tc12	Ryser, J., Glinz, M.	Using Dependency Charts to Improve Scenario-Based Testing [Ryser, 2000]	Conferencia	2000	Busqueda secundaria. Referencia de [Some, 2007]
Tc13	Van Den Berg K.G., Simons A.J.H.	Control-flow semantics of use cases in UML [Van Den Berg, 1999]	Journal	1999	Búsqueda primaria

Las técnicas se evaluaron empleando los siguientes criterios: utilización de casos de uso o técnica equivalente (ver subsección 2.8.1), empleo de dependencias entre requisitos como precondiciones (ver subsección 2.8.2) o secuencias entre requisitos (ver subsección 2.8.3), y uso de notación gráfica o empleo de diagramas de UML (ver subsección 2.8.4).

2.8.1 Utilización de casos de uso o técnica equivalente

La diferencia entre escenarios y casos de uso radica en que un escenario es una instancia de un caso de uso [Bruegge, 2002]. El caso de uso es el que especifica todos los escenarios posibles para una parte de funcionalidad dada; es decir, todos los escenarios similares se agrupan en un solo caso de uso. Tomando en cuenta esta diferencia, se realizó una comparación entre las técnicas. También se tomó en cuenta que hay técnicas que no emplean casos de uso, pero son equivalentes a ellas (no equivalentes a escenarios) como el caso de los *user stories*, historias de usuario en castellano, *features* o simplemente requisitos.

La Tabla 2-32 muestra qué técnicas emplean casos de uso y cuáles son equivalentes. La primera columna indica el código que se ha asignado a la técnica; la segunda columna, los autores; la tercera columna, si la técnica emplea casos de uso; y la última columna muestra qué equivalencia a los casos de uso utiliza la técnica solo si no utiliza casos de uso.

Tabla 2-32. Técnicas que emplean casos de uso o equivalente

Código	Autores	Emplea casos de uso	Emplea equivalente a casos de uso
Tc1	Gomez, A., et. al	No	Sí, user stories
Tc2	Sapna, P.G., et. al	No	No, escenarios de casos de uso
Tc3	Balikuddembe J.K., et. al	Sí	-
Tc4	Szoke, A.	Sí	-
Tc5	Some S.S.	Sí	-
Tc6	Kundu D., et.al	No	No, escenarios de casos de uso
Tc7	Whittle, J.,	Sí	-
Tc8	Zhang, W., et. al	No	Sí, feature
Tc9	Denne, M., et.al	No	Sí, requisitos
Tc10	Carlshamre, P.	No	Sí, requisitos
	Carlshamre P., et. al		
Tc11	Moisiadis F.	No	No, escenarios de casos de uso
Tc12	Ryser, J., et. al	Sí	-
Tc13	Van Den Berg K.G., et. al	Sí	-

Tal y como se observa en la tabla anterior (Tabla 2-32), casi la mitad de las técnicas emplean casos de uso. Aquellas técnicas que no emplean casos de uso

usan como equivalente a los *user stories* (técnica Tc1), *features* (técnica Tc8) o simplemente requisitos (técnicas Tc9 y Tc10). Cabe resaltar que hay técnicas que han empleado escenarios de casos de uso, pero no se han considerado como equivalentes a ellos; porque, como se indicó al inicio de esta subsección, son dos conceptos relacionados, pero diferentes.

2.8.2 Precondiciones

El concepto de precondición está definido explícitamente en los casos de uso [Jacobson, 1992] [OMG, 2006]. Las precondiciones detallan los estados en los que debe estar el sistema para que se pueda ejecutar el caso de uso; por ejemplo: para que un actor pueda ejecutar un caso de uso específico previamente ha debido validarse en el sistema (ejecución de un escenario del caso de uso *login*). Cabe resaltar que el concepto de precondición también puede ser aplicable a requisitos sin importar cómo ha sido representado: *user stories*, *features*, etc. La precondición podría ser utilizada para definir la priorización de casos de uso o equivalente y define un tipo de precedencia o dependencia entre casos de uso.

Solo cinco técnicas consideran las precondiciones como un elemento a tomar en cuenta para la priorización de requisitos y casos de uso. Esas técnicas son la técnica Tc5 de Some [Some, 2007], Tc6 de Kundu et. al [Kundu, 2007], Tc8 de Zhang et. al [Zhang, 2001], Tc10 de Carlshamre et. al [Carlshamre, 2002] y Tc12 de Ryser et. al [Ryser, 2000]. De estas cinco técnicas, la Tc10 propuesta por Carlshamre no indica que emplea precondiciones, pero sí considera una relación entre requisitos que sería equivalente, la cual denomina como “*REQUIRE*”, requerir en castellano.

2.8.3 Secuencias entre requisitos

Según Somé [Some, 2007], existe una restricción de secuencia en un caso de uso CU1 debido a un caso de uso CU2, cuando una instancia de CU2 se puede ejecutar solo si una instancia de CU1 previamente se ha ejecutado. Por ejemplo, para los casos de uso “Procesar ofertas de proveedores” y “Registrar ofertas de proveedores”, primero debería ejecutarse instancias de “Registrar ofertas de proveedores” antes de “Procesar ofertas de proveedores”, esto quiere decir que este último caso de uso tendría una restricción de secuencia con “Registrar ofertas”.

Las restricciones de secuencia podrían confundirse con las precondiciones, pero no son las mismas. Siguiendo el ejemplo mostrado en el párrafo anterior, el usuario sí podría ejecutar la funcionalidad del sistema “Procesar ofertas de proveedores” sin haber seleccionado previamente “Registrar ofertas” ni haber registrado ofertas. En este caso, el sistema mostraría una lista vacía de ofertas a procesar o mostraría ofertas relacionadas a otros productos. Si las instancias de “Registrar ofertas” fueran precondiciones de “Procesar ofertas de proveedores” entonces el usuario estaría impedido de ejecutar la funcionalidad para procesar ofertas de proveedores si es que previamente no se ha registrado ninguna oferta.

Al revisar las técnicas siguiendo el criterio de restricciones de secuencia, se pudieron clasificar en dos grupos: 1) técnicas que no consideran esta restricción o que no definen ningún tipo de regla para determinarlas y 2) técnicas que definen reglas para determinar estas restricciones o utiliza algún diagrama, por ejemplo el diagrama de actividades, que obliga a tomar en cuenta este tipo de restricciones.

Las técnicas que pertenecen al primer grupo, es decir aquellas no toman en cuenta estas restricciones o no definen reglas, son las técnicas Tc3 de Balikuddembe et. al [Balikuddembe, 2008], Tc4 de Szoke [Szoke, 2008], Tc9 de Denne et. al [Denne, 2004] y Tc13 de Van Den Berg et. al [Van Den Berg, 1999]. Cabe resaltar que la técnica Tc9 de Denne et. al indica ejemplos que tendrían relación a las restricciones de secuencia, pero no define reglas. En el caso de la técnica Tc13 de Van Den Berg et. al, muestra una relación de precedencias entre casos de uso, pero no indica reglas para determinar esa precedencia.

En cuanto a aquellas técnicas que emplean algún diagrama que de alguna manera considera este tipo de restricción, se tienen a las técnicas Tc2 de Sapna et. al [Sapna, 2009], Tc7 de Whittle [Whittle, 2007] y Tc11 de Mosiadis [Mosiadis, 2000] que emplean diagramas de actividades; la técnica T6 de Kundu et. al [Kundu, 2007] que utiliza diagramas de secuencia; y la técnica Tc1 de Gomez et. al [Gomez, 2010] (técnica Tc1) que muestra un ejemplo que incluye un diagrama relacionado a este tipo de restricción entre historias de usuario.

Sobre las técnicas que definen reglas asociadas a este tipo de restricción, se tiene a la técnica Tc5 de Somé [Some, 2007] que define las relaciones “*follow*” y “*enable*” entre casos de uso; Tc8 de Zhang et. al [Zhang, 2005] que especifica las relaciones “*inform*” y “*flow*” entre requisitos; Tc10 de Carlshamre [Carlshamre, 2002] que define las relaciones “*requires*” y “*temporal*” entre requisitos; y, finalmente, Tc12 de Ryser [Ryser, 2000] que determina diferentes relaciones entre casos de uso asociadas a esta restricción y propone diagramas propios que permiten representarlas gráficamente.

2.8.4 Notación gráfica y notación UML

Algunas de las técnicas emplean diagramas para representar la dependencia entre casos de uso y otras se basan en los diagramas especificados en UML. Tomando estos dos criterios, la Tabla 2-33 muestra en la penúltima columna si emplea o no una notación gráfica y en la última columna, si utiliza notación UML. En esta última columna se indica con un guión (“-”) si la técnica no emplea diagramas para representar la dependencia entre casos de uso o requisitos.

Tabla 2-33. Técnicas que emplean notación gráfica y UML

Código	Autores	Notación Gráfica	Notación UML
Tc1	Gomez, A., et. al	Sí	No
Tc2	Sapna, P.G., et .al	Sí	Sí, diagrama de actividades
Tc3	Balikuddembe J.K., et. al	No	-
Tc4	Szoke, A.	Sí	Sí, casos de uso
Tc5	Some S.S.	No	-
Tc6	Kundu D., et.al	Sí	Sí, diagrama de secuencias
Tc7	Whittle, J.,	Sí	Sí, diagrama de actividades
Tc8	Zhang, W., et. al	Sí	Sí, diagramas de clases
Tc9	Denne, M., et.al	No	-
Tc10	Carlshamre, P.	No	-
	Carlshamre P., et. al		
Tc11	Moisiadis F.	Sí	Sí, diagrama de actividades
Tc12	Ryser, J., et. al	Sí	No
Tc13	Van Den Berg K.G., et. al	Sí	Sí, casos de uso

Según lo mostrado en la Tabla 2-33, solo cuatro técnicas (Tc3, Tc5, Tc9 y Tc10) no emplean diagramas para representar la dependencia entre requisitos. De las que sí emplean diagramas, solo dos no emplean ningún diagrama de UML (Tc1 y Tc12). Las que sí emplean diagramas UML, utilizan diagramas de actividades (Tc2, Tc7 y Tc11), diagramas de secuencias (Tc6), diagramas de clases (Tc8) y diagramas de casos de uso al que le añaden más relaciones entre casos de uso de las que define UML (Tc4 y Tc15).

2.8.5 Conclusiones de la revisión de técnicas para priorizar la secuencia de construcción de casos de uso

Como se ha podido observar al revisar los artículos seleccionados, ninguno de ellos toma en cuenta la facilidad de construcción de requisitos según el punto de vista del desarrollador. Por lo general, se realiza la priorización de la construcción teniendo como base las preferencias del cliente y de los usuarios.

Tomando en cuenta las recomendaciones de Webster y Watson [Webster, 2002] para sintetizar la comparación entre propuestas o conceptos, se realizó la Tabla 2-34 que muestra un cuadro comparativo resumen de las técnicas que se evaluaron. La primera columna de la Tabla 2-29 muestra el código de la técnica y la segunda columna, sus autores. En la tercera columna se indica si la técnica emplea casos de uso para priorizar la construcción; en la cuarta columna, si la técnica toma en cuenta las precondiciones; en la quinta columna, si definen reglas para las restricciones de secuencia; y, finalmente, en las dos últimas columnas, si se emplea alguna notación gráfica y si utiliza UML. En todas las columnas se ha indicado con una equis ("X"), si la técnica cumple con el criterio definido y se deja la celda en blanco si no lo cumple.

Tabla 2-34. Tabla comparativa entre técnicas para definir secuencia de construcción de casos de uso

Código	Autores	Emplea casos de uso	Considera precondiciones o equivalente	Reglas para restricciones de secuencia	Notación Gráfica	Notación UML
Tc1	Gomez, A., et. al			X	X	
Tc2	Sapna, P.G., et. al			X	X	X
Tc3	Balikuddembe J.K., et. al	X				
Tc4	Szoke, A.	X			X	X
Tc5	Some S.S.	X	X	X		
Tc6	Kundu D., et.al		X	X	X	X
Tc7	Whittle, J.,	X		X	X	X
Tc8	Zhang, W., et. al		X	X	X	X
Tc9	Denne, M., et.al					
Tc10	Carlshamre P., et. al		X	X		
Tc11	Moisiadis F.			X	X	X
Tc12	Ryser, J., et. al	X	X	X	X	
Tc13	Van Den Berg K.G., et. al	X			X	X

Como se puede observar en la Tabla 2-34, ninguna de las técnicas cumplen con todos los criterios que se han establecido. Sin embargo, la técnica que cumple casi con todos los criterios es la técnica Tc12 de Ryser et. al [Ryser, 2000], a la cual le faltaría emplear notación UML. También, la técnica Tc8 de Zhang et. al [Zhang, 2005] cumple con casi todos los criterios, pero no emplea casos de uso.

Por lo tanto, debería proponerse una nueva técnica que permita definir la secuencia de construcción de casos de uso que cumpla con lo siguiente: considere las precondiciones, defina reglas que permitan determinar las restricciones entre secuencias de construcción de casos de uso y utilice UML como notación.

2.9 Amenazas a la Validez de la Revisión Sistemática

La discusión de las amenazas de validez es importante para poder juzgar las fortalezas y limitaciones del estudio en relación a la validez de los resultados obtenidos. Las principales amenazas a la validez de esta revisión sistemática están relacionadas con las bases de datos empleadas, la búsqueda de la llamada “literatura gris”, la parcialidad en la selección de los artículos, la exclusión de artículos relevantes y los términos empleados en las búsquedas.

2.9.1 Bases de datos empleadas y “literatura gris”

El empleo de bases de datos como Scopus, ISI Web of Science o IEEE Xplore para la búsqueda primaria significa que se pudieron haber omitido artículos en revistas y conferencias menos conocidas. En relación a este punto, tal vez el mayor problema, en relación a la validez de esta revisión sistemática, es si se ha fallado en encontrar estudios primarios relevantes; aunque las bases de datos seleccionadas nos aseguran que el campo de búsqueda de artículos en

conferencias y *journals* es lo bastante amplio para lograr nuestro propósito. No obstante, hay que tener en cuenta que es imposible lograr una total completitud en el campo de estudio; porque podrían existir documentos relacionados al tema que no han sido incluidos.

Por ello, como medida de reducción de esta amenaza, se empleó Scholar Google para encontrar las citaciones de los artículos que fueron seleccionados durante las búsquedas primarias. Scholar Google indexa artículos que se encuentran en las bases de datos seleccionadas y artículos de revistas y conferencias menos conocidas. No se empleó esta base de datos en las búsqueda primarias, debido a que no permite la utilización de cadenas complejas.

La denominada “literatura gris” no fue incluida en la búsqueda primaria, porque es difícil de identificar. Aunque se emplearon las bases de datos recomendadas por muchas revisiones sistemáticas en Ingeniería de Software, la inclusión de “literatura gris” podría incrementar la validez del presente estudio. Sin embargo, el empleo de Scholar Google en la búsqueda secundaria permitió averiguar la existencia de parte de este tipo de literatura relacionada al tema.

2.9.2 Parcialidad en la selección de artículos

Debido a que solo una persona llevó a cabo la búsqueda, selección y análisis de los estudios, podría existir una amenaza en relación a la parcialidad del investigador. Esto se mitigó de la siguiente manera: el protocolo de esta revisión sistemática fue evaluada y revisada por dos investigadores y la claridad de los criterios de inclusión/exclusión fue probada con un grupo de artículos que previamente se habían seleccionado. Además, estos criterios fueron definidos de manera que puedan ser contestados con un simple sí o no.

2.9.3 Exclusión de artículos relevantes

El riesgo de excluir artículos relevantes se redujo al incluir en una revisión más detallada a aquellos artículos en los cuales se tenían dudas durante la revisión de los resúmenes o *abstracts*. Aunque en las guías de Kitchenham [Kitchenham, 2007] se sugiere el empleo de procedimientos para evaluar la calidad de los artículos, de manera que se puedan descartar aquellos que no sigan ciertos estándares; para esta revisión no se hizo este tipo de evaluación, porque se quiso analizar la mayor cantidad de propuestas. Al respecto, Brereton et. al [Brereton, 2005] hace notar que en las revisiones sistemáticas que se realizan en Medicina se enfatiza la evaluación de calidad en las búsquedas primarias; sin embargo, esto depende del tipo de revisión que se esté llevando a cabo. Para nuestro caso, el objetivo principal era conocer qué propuestas existen hasta la actualidad sobre técnicas para estimar y planificar el esfuerzo en desarrollos de software orientado a objetos y que siga un modelo de ciclo de vida incremental; por ello la calidad de los estudios no fue considerado ni importante ni crítico.

2.9.4 Términos empleados en las búsquedas

Los términos empleados para construir la cadena de búsqueda tienen más sinónimos; por lo que no se puede garantizar que todo el trabajo relacionado al alcance de la revisión de literatura haya sido encontrado mediante las consultas en las bases de datos. Pero a pesar de ello, para la definición de términos, no solo se consideraron algunos sinónimos de los diccionarios, sino también los términos que fueron empleados por otros autores que han realizado revisiones sistemáticas en el área.

2.10 Conclusiones del Estado de la Cuestión

Las conclusiones obtenidas en el estudio del estado de la cuestión se han dividido en cuatro partes. En la primera de ellas, se muestran las conclusiones relacionadas al proceso de búsqueda y selección de la revisión sistemática (ver subsección 2.10.1); en la segunda, las conclusiones de la revisión de técnicas que hacen el cálculo de PF para modelos orientados a objetos (ver subsección 2.10.2); en la tercera, las conclusiones de la revisión de técnicas que estiman el esfuerzo en proyectos que siguen el modelo de ciclo de vida incremental (ver subsección 2.10.3); y, en la última, las conclusiones de la revisión de las técnicas que permiten priorizar la construcción de casos de uso (ver subsección 2.10.4).

2.10.1 Búsqueda y selección de artículos

El estudio del estado de la cuestión se realizó mediante la denominada “Revisión Sistemática”. Este tipo de revisiones, a diferencia de las revisiones de literatura tradicionales, emplean métodos científicos que minimizan la parcialidad y los errores [Mulrow, 1998].

Inicialmente, se determinaron las preguntas de investigación mediante el criterio PICOC sugerido por Petticrew y Roberts [Petticrew, 2006]. A continuación, se determinaron las cadenas de búsqueda que fueron utilizadas en cinco bases de datos: Scopus, ISI Web of Science, Science Direct, IEEE Xplore, ACM Digital Library, Computer Database y Engineering Village (Inspec & Compendex). Lamentablemente, se obtuvieron pocos resultados en las búsquedas y no se seleccionó ningún artículo. Esto quiere decir que **no se encontraron técnicas, métodos o aproximaciones que permita estimar y planificar proyectos de software orientado a objetos que empleen ciclos de vida incremental y Puntos de función**. Por ello, se tuvo que redefinir la estrategia de búsqueda y selección.

En la redefinición de la estrategia de búsqueda se tomaron en cuenta las tres subpreguntas de investigación que se definieron inicialmente. Las subpreguntas fueron las siguientes: **2(a)** ¿Qué técnicas o modelos de conversión se han definido para calcular puntos de función empleando modelos orientados a objetos?, **2(b)** ¿Qué actividades/tareas/pasos se han definido para calcular el esfuerzo requerido para cada incremento? y **2(c)** ¿Qué

actividades/tareas/pasos se han definido para determinar la prioridad en la construcción de casos de uso?

Finalmente, por cada subpregunta, se tuvo que hacer búsquedas en las bases de datos, definir criterios de inclusión/exclusión y seleccionar artículos que permitieran responder a cada una de esas subpreguntas.

Durante las búsquedas se pudo observar que la base de datos Scopus fue la que permitió encontrar la mayor cantidad de artículos que finalmente fueron seleccionados. En las búsquedas relacionadas a las subpreguntas 2(a) y 2(b), Scopus fue la única base de datos que encontró los artículos que finalmente fueron seleccionados. No obstante, en la búsqueda de la subpregunta 2(c) no ocurrió lo mismo, pero Scopus sí fue la que permitió obtener la mayor cantidad de artículos seleccionados a diferencia de las demás bases de datos. Por ello, no se puede afirmar que sería suficiente si solo se utilizara Scopus. Por otro lado, Computer Database fue la base de datos con la que se obtuvo menos de un artículo seleccionado.

2.10.2 Cálculo de Puntos de función en modelos orientados a objetos

La revisión de artículos que definen reglas para calcular PF permitió observar que muchas de ellas no consideran lo especificado en los manuales de Puntos de función (PF) definidos por el Ifpug; algunas sí consideran las reglas especificadas por el Ifpug, pero no toman en cuenta los últimos manuales [Ifpug, 2004][Ifpug, 2009]. Solo una técnica toma como base uno de los últimos manuales de Puntos de función [Lavazza, 2008].

La revisión a detalle de las técnicas consideró la evaluación de las reglas que ellas proponen para el cálculo de PF para los ficheros (denominada originalmente como *Data Functions*) y para el cálculo de PF para las transacciones (denominada originalmente como *Transactional Functions*).

Aunque todas las técnicas utilizan diagramas de clases para realizar el cálculo de ficheros de Puntos de función, se pudo determinar que las técnicas no definen reglas para las relaciones que pueden existir entre clases (asociación, agregación, composición, herencia y clase asociación) y que cumplan con las especificaciones de los últimos manuales de Puntos de función [Ifpug, 2004] [Ifpug, 2009].

En cuanto a las transacciones, la mayoría de las técnicas cumple con casi todo lo especificado en los manuales de Puntos de función. No obstante, no existe consenso en indicar qué diagrama de UML se debe utilizar para realizar este tipo de cálculo y es muy dependiente de la metodología de desarrollo orientado a objetos que la técnica toma en cuenta.

Considerando lo analizado en la revisión de este tipo de técnicas, se puede concluir que no existen técnicas que estén de acuerdo a todos los criterios siguientes:

- Que definan reglas para calcular ficheros que consideren todas las relaciones que pueden existir entre clases: asociación, agregación, composición, herencia y clase asociación.
- Que sean independientes de las metodologías de desarrollo de software orientado a objetos para el cálculo de las transacciones de PF. Además, que no añadan diagramas que solo sean útiles para el cálculo de puntos de función, pero que no puedan posteriormente ser empleados en etapas siguientes del desarrollo, por ejemplo el diseño.
- Que cumplan con lo que especifican los últimos manuales de PF definidos por el Ifpug, tanto para el cálculo de ficheros como para el cálculo de transacciones.

2.10.3 Cálculo del esfuerzo requerido por cada incremento

Debido a la experiencia obtenida en la búsqueda inicial en la que no se obtuvo ningún resultado relacionado al modelo de ciclo de vida incremental, pero sí uno relacionado al modelo de ciclo de vida iterativo y dado que ambos modelos de ciclo de vida tienen elementos en común, en los criterios de inclusión/exclusión se consideraron ambos tipos de modelos.

Al revisar las tres técnicas para determinar el esfuerzo requerido para cada incremento o iteración que finalmente fueron seleccionadas, se pudo observar que ninguna de ellas define reglas para poder priorizar la construcción de casos de uso. Por ello, se tuvo que realizar una nueva búsqueda para encontrar técnicas que indiquen como realizar esta tarea y así poder responder la subpregunta de investigación 2(c).

En cuanto a la estimación de incrementos, ninguna de las técnicas define cómo dividir los PF calculados entre iteraciones o incrementos, siendo el más crítico la división de PF correspondientes a ficheros. Además, la productividad entre iteraciones e incrementos podría aumentar o disminuir por diversos factores como rotación de personal, incremento en la experiencia para desarrollar software, incremento en el conocimiento de las herramientas y lenguajes con el transcurrir del tiempo, etc. No obstante, las técnicas no consideran los cambios de productividad del equipo de desarrollo.

Tomando en cuenta lo analizado en la revisión de este tipo de técnicas, se puede concluir que no existen técnicas que estén de acuerdo a todos los criterios siguientes:

- Que definan reglas para dividir los puntos de función del software a construir entre los incrementos.
- Que consideren el cambio de productividad del equipo de desarrollo para estimar el esfuerzo de construcción de los incrementos.

- Que definan la priorización de la construcción de incrementos; es decir, determinar que caso de uso debería construirse en qué incremento.

2.10.4 Definición de la priorización o secuencia de construcción de casos de uso

Como ninguna de las tres técnicas encontradas en la evaluación de la estimación del esfuerzo para modelos de ciclo de vida incremental o iterativo definen reglas para poder priorizar o definir la secuencia de construcción de casos de uso, se tuvo que hacer una búsqueda de artículos sin importar que modelo de ciclo de vida se sigue. El objetivo de esta búsqueda fue encontrar artículos que indicaran cómo definir la secuencia de construcción de casos de uso.

Se pudo observar que ninguna de las técnicas considera la facilidad de construcción de requisitos según el punto de vista del desarrollador. Por lo general, se realiza la priorización de la construcción teniendo como base las preferencias del cliente y de los usuarios, y en algunos casos, de manera adicional, la del desarrollador.

Aunque realizar la priorización teniendo en cuenta las precedencias o dependencias entre requisitos podría facilitarle el trabajo del desarrollador, muy pocas las consideran. Por ello, se analizó si las técnicas consideran dos tipos de dependencias: precondiciones y secuencias entre requisitos. También, si utilizan algún tipo de diagrama para mostrar las dependencias o precedencias y si emplean algún diagrama de UML.

En cuanto la evaluación de precedencia o dependencia entre requisitos, ninguna de las técnicas cumple con todos los criterios que se definieron: uso de precondiciones, definición de secuencias entre requisitos y empleo de diagramas. Sin embargo, la técnica Ryser et. al [Ryser, 2000] cumple con casi todos los criterios, pero no emplea notación UML. También, la técnica de Zhang et. al [Zhang, 2005] cumple con casi todos los criterios, pero no emplea casos de uso y representa los requisitos como diagramas de clases en su artículo.

Tomando en cuenta lo analizado en la revisión de este tipo de técnicas, se puede concluir que no existen técnicas que estén de acuerdo a todos los criterios siguientes:

- Que consideren la facilidad de construcción de casos de uso según el punto de vista del desarrollador.
- Que tomen en cuenta las precedencias o dependencias entre casos de uso: precondiciones y secuencias.
- Que empleen notación gráfica y diagramas de UML.

Planteamiento del Problema

Capítulo

3

El objetivo del capítulo es presentar el enfoque del problema de investigación, el cual se ha definido luego de haber realizado el estudio del estado de la cuestión, qué hipótesis de trabajo se van a considerar y la descripción de la solución propuesta.

Inicialmente, en la Sección 3.1, se define el problema tomando en cuenta las conclusiones obtenidas en el estado de la cuestión. A continuación, en la Sección 3.2, se definen las hipótesis de trabajo; y en la Sección 3.3, se da una visión general de la solución propuesta. Finalmente, se detalla en la Sección 3.4 la estructura de la resolución que se incluyen en los capítulos siguientes.

3.1 Definición del Problema

La simplicidad del modelo de ciclo en cascada hace que sea fácil explicarlo a los clientes que no están familiarizados con el desarrollo de software [Pfleeger, 2002]. Sin embargo, en la literatura, se puede encontrar muchas discusiones sobre los numerosos problemas asociados con este modelo. La Figura 3-1 ilustra un ejemplo presentado por Jaaksi [Jaaksi, 1999] de un proyecto típico que emplea un modelo de ciclo de vida en cascada y que finalmente sufre un retraso en la entrega del software.

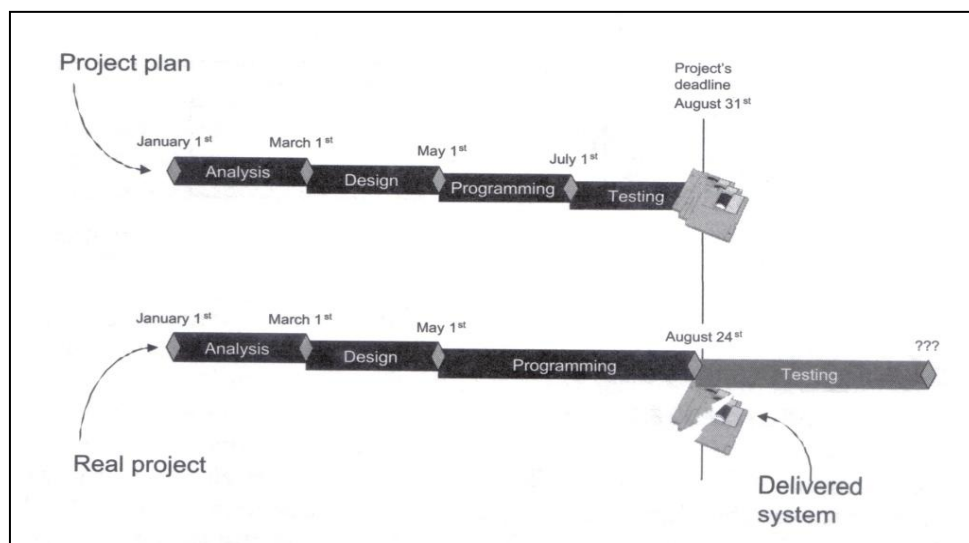


Figura 3-1. Comparación entre lo planificado y la ejecución de un proyecto en cascada (tomado de [Jaaksi, 1999])

Según el plan presentado en la Figura 3-1, cada fase dura dos meses, el proyecto se inicia el primero de enero y debería finalizar el 31 de agosto. Según lo observado, el análisis y el diseño se producen sin ningún contratiempo. Sin embargo, los programadores encuentran problemas críticos en la fase de programación y se demoran dos veces del tiempo planeado. Como en la mayoría de casos, se tiene una fecha fija de entrega y un presupuesto definido; por lo que la programación no se puede extender ni se pueden contratar a más programadores. Para la fecha de entrega, la calidad del software es incierta, porque no ha pasado por la fase de pruebas.

Si se hubiera seguido un modelo de ciclo de vida incremental, el escenario podría haber sido diferente. La Figura 3-2 muestra el mismo proyecto con este tipo de modelo de ciclo de vida. Para el proyecto, se han definido tres incrementos: el primero de ellos implementa los casos de uso 1 y 2; el segundo, los casos 3 y 4; y, finalmente, el último, los casos 5 y 6.

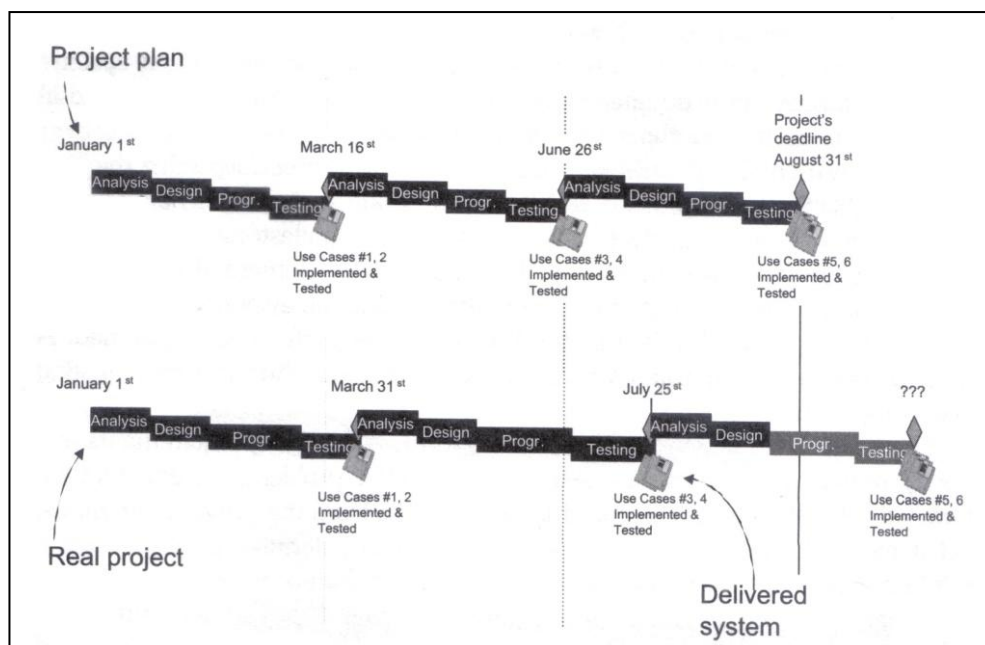


Figura 3-2. Comparación entre lo planificado y la ejecución de un proyecto incremental (tomado de [Jaaksi, 1999])

Aunque empleando el modelo de ciclo de vida incremental que se muestra en la Figura 3-2 también se produce un retraso en la entrega final del producto; a diferencia del proyecto que sigue el modelo de ciclo de vida en cascada, el jefe de proyecto puede informar exactamente qué funcionalidad está disponible para que pueda ser utilizada por los usuarios y cuál faltaría. Esto quiere decir que un modelo de ciclo de vida incremental no reduce el riesgo de entregar el software completo a tiempo, pero sí asegura que se puedan entregar funcionalidades del software que sí se puedan utilizar.

El desarrollo de software incremental, que es una parte fundamental en los enfoques ágiles, es mejor que un desarrollo en cascada para la mayor parte de los sistemas empresariales actuales [Sommerville, 2011]. Además, el desarrollo incremental refleja la forma en que se resuelven los problemas; porque rara vez se trabaja por adelantado una solución completa de un problema y más bien se avanza en una serie de pasos hacia una solución y se retrocede cuando se detecta que se han cometido errores.

Lamentablemente, del estudio del estado de la cuestión se concluye que no existen trabajos suficientes que realicen la adaptación de la técnica de Puntos de función para ciclos de vida incremental y el paradigma orientado a objetos. Aunque existen técnicas que permiten hacer el cálculo de PF empleando diagramas orientados a objetos, las técnicas propuestas no cumplen con las reglas definidas en los últimos manuales de Puntos de función [Ifpug, 2004] [Ifpug, 2009].

A pesar de que muchos autores sugieren el empleo del modelo de ciclo de vida incremental en vez al modelo de ciclo de vida en cascada debido a sus múltiples ventajas; no existen propuestas para utilizar la técnica de Puntos de función con este tipo de modelos.

En particular, el estudio del estado de la cuestión ha arrojado las siguientes conclusiones:

- No existen técnicas que definan reglas para calcular PF considerando todas las relaciones que pueden existir entre clases (asociación, agregación, composición, herencia y clase asociación), que sean independientes de las metodologías de desarrollo de software orientado a objetos y que cumplan con lo que especifican los últimos manuales de Puntos de función del Ifpug [Ifpug, 2004][Ifpug, 2009].
- No existen técnicas que definan reglas para dividir los PF de un software entre incrementos a desarrollar y que consideren el cambio de productividad del equipo de desarrollo para estimar el esfuerzo para construir los incrementos.
- No existen técnicas que consideren la facilidad de construcción de casos de uso según el punto de vista del desarrollador, que tomen en cuenta las dependencias entre casos de uso (precondiciones y secuencias) y que empleen diagramas de UML.

3.2 Hipótesis de Trabajo

El objetivo de esta investigación es la creación de un conjunto de técnicas que permita realizar la estimación y planificación basada en Puntos de función para desarrollos de software orientado a objetos y que sigan un modelo de ciclo de vida incremental.

Para el desarrollo de este conjunto de técnicas, se ha definido una hipótesis general que se ha dividido en tres sub-hipótesis de trabajo. La hipótesis general se ha definido de la siguiente manera:

- HG:** Es posible definir un conjunto de técnicas que permitan apoyar la estimación y planificación de proyectos de desarrollo de software que empleen modelos de ciclos de vida incremental, paradigma orientado a objetos y Puntos de función.

Esta hipótesis general, compleja para ser validada en su conjunto, se descompone en las siguientes sub-hipótesis:

- SH1. Las técnicas propuestas permiten realizar el cálculo de puntos de función empleando diagramas del paradigma orientado a objetos sin contradecir las especificaciones de los últimos manuales de Puntos de función.**

Las técnicas deberán cumplir con las reglas que se incluyen en los últimos manuales de Puntos de función [Ifpug, 2004] [Ifpug, 2009] y deberán incluir las relaciones entre clases: asociación, agregación, composición, herencia y clase asociación. Además, las reglas definidas para el cálculo de ficheros con diagramas de clases deberían dar resultados por lo menos igual de precisos que si se utilizaran diagramas Entidad-Relación que es lo que propone la técnica original del Ifpug. A nivel de transacciones de Puntos de función las técnicas deberán ser independientes de las metodologías de desarrollo de software.

- SH2. Las técnicas permiten determinar la priorización de los casos de uso a construir de manera más efectiva que las técnicas ad hoc o propias de los desarrolladores.**

Las técnicas deberán considerar la facilidad de construcción de casos de uso según el punto de vista del desarrollador y también deberán tomar en cuenta las precedencias o dependencias entre casos de uso como las precondiciones y las secuencias entre casos de uso. También, deberá emplear notación UML. Las técnicas deberán ser más efectivas que las técnicas que utilizan actualmente los desarrolladores para definir la secuencia o priorizar los casos de uso a construir.

- SH3. Las técnicas tienen la capacidad de estimar el esfuerzo requerido para construir un incremento con una diferencia menor o igual al 20% entre el esfuerzo estimado y el real.**

Las técnicas, para lograr esa diferencia, deberían considerar los cambios de productividad del equipo de desarrollo entre incrementos. Además, tiene que indicar explícitamente como dividir los puntos de función calculados para todo el proyecto entre los

incrementos. Finalmente, la diferencia entre el esfuerzo real estimado y el real no debería exceder del 20%.

3.3 Descripción de la Solución Propuesta

La Figura 3-3 muestra una visión general de Tupuy y de las técnicas que están compuestas. Para que se pueda trabajar con Tupuy, se necesita que previamente se hayan realizado los diagramas de clases de análisis y las especificaciones de casos de uso. Las técnicas que compone Tupuy son tres: *Use Case Precedence Diagram* (UCPD), UML₂FP e Incremental-FP. Estas se muestran en la figura como rectángulos de color gris.

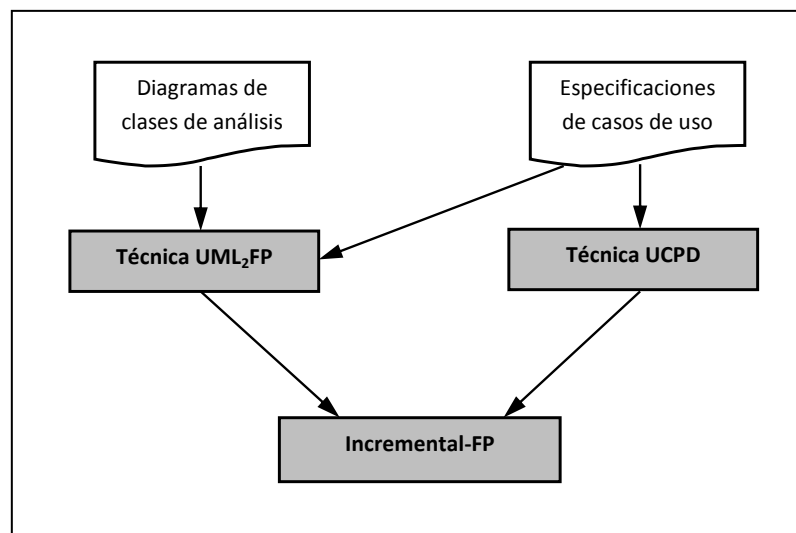


Figura 3-3. Esquema general de Tupuy

La **técnica UML₂FP** permite realizar el cálculo de PFSA. Para ello, emplea el diagrama de clases de análisis [Jaaksi, 1999], el cual también es conocido en otras metodologías como modelo de dominio [Larman, 1999]; también, utiliza las especificaciones de caso de uso. Para la **técnica UCPD**, se emplea la especificación de los casos de uso para definir la secuencia de construcción de casos de uso. Finalmente, la **técnica Incremental-FP** utiliza los resultados obtenidos de emplear las técnicas UCPD y UML₂FP. Incremental-FP comprende la definición de los incrementos a construir y la determinación de qué casos de uso se va a desarrollar en cada incremento. Además, se estima el esfuerzo que se requiere para concluir cada incremento.

Cada una de las técnicas que compone Tupuy responde a cada una de las sub-hipótesis planteadas. La **técnica UML₂FP** se encarga de resolver lo planteado en la SH1 “Las técnicas propuestas permiten realizar el cálculo de puntos de función empleando diagramas del paradigma orientado a objetos sin

contradecir las especificaciones de los últimos manuales de Puntos de función”; la **técnica UCPD** se relaciona a la SH2 “Las técnicas permiten determinar la priorización de los casos de uso a construir de manera más efectiva que las técnicas ad hoc o propias de los desarrolladores”; y, finalmente, la **técnica Incremental-FP** responde a la SH3: “Las técnicas tienen la capacidad de estimar el esfuerzo requerido para construir un incremento con una diferencia menor o igual al 20% entre el esfuerzo estimado y el real”.

Finalmente, Tupuy, que está definida como el conjunto de las tres técnicas mencionadas anteriormente, responde a la hipótesis general “Es posible definir un conjunto de técnicas que permitan apoyar la estimación y planificación de proyectos de desarrollo de software que empleen modelos ciclos de vida incremental, paradigma orientado a objetos y Puntos de función” y, por supuesto, a todas las sub-hipótesis planteadas.

3.4 Estructura de la Resolución

En los siguientes capítulos se detalla el conjunto de técnicas propuestas y las evaluaciones que se han realizado. El capítulo 4 detalla al conjunto de técnicas que se proponen y que se ha denominado como Tupuy. Adicionalmente, antes de explicar el conjunto de técnicas, se incluyen los conceptos que se han considerado para definir a Tupuy.

En el capítulo 5, se detalla la validación de la identificación de ficheros y sus RET de la técnica UML₂FP, la que se ha realizado mediante experimentos controlados con profesionales, alumnos de pregrado y alumnos de posgrado.

Luego, en el capítulo 6 se incluye la evaluación que se ha hecho a una de las técnicas que conforman Tupuy: *Use Case Precedence Diagram* (UCPD). La evaluación se ha realizado mediante experimentos controlados con profesionales, alumnos de pregrado y alumnos de posgrado para determinar si la técnica es más efectiva que las técnicas ad hoc o propias cuando se quiere determinar la secuencia de construcción de casos de uso.

Finalmente, en el capítulo 7 se incluye la evaluación de todas las técnicas, pero principalmente de la técnica Incremental-FP. Esta evaluación se ha realizado mediante proyectos con alumnos de pregrado que emplearon la técnica. También, se han utilizado los datos de proyectos con alumnos para hacer simulaciones, cuando estos alumnos no utilizaron la técnica.

Aproximación de la Solución: Tupuy

Capítulo

4

El objetivo del capítulo es definir un conjunto de técnicas que permita realizar la planificación y estimación para el desarrollo de software que sigue el modelo de ciclo de vida incremental. Al conjunto de técnicas se le ha denominado Tupuy. La palabra Tupuy proviene de la lengua quechua, idioma de los incas, que significa “medir” o “pesar”.

El capítulo se estructura en tres secciones. La Sección 4.1 muestra los conceptos en los que se fundamenta Tupuy: la especificación de casos de uso (subsección 4.1.1), los diagramas de clases de análisis (subsección 4.1.2) y la estimación del esfuerzo con Cocomo II y Puntos de Función (subsección 4.1.3). La Sección 4.2 describe Tupuy con las tres técnicas que incluye: UML₂FP (subsección 4.2.1), UCPD (subsección 4.2.2) e Incremental-FP (subsección 4.2.3). También se incluye un ejemplo de aplicación de Tupuy (Sección 4.3). Finalmente, en la Sección 4.4 se indica como se ha realizado la evaluación de Tupuy, la cual se incluye en los capítulos 5, 6 y 7.

4.1 Conceptos en los que se Fundamenta la Aproximación a la Solución

Tupuy se fundamenta en el paradigma orientado a objetos. Por ello, esta sección muestra una revisión de los conceptos en los que se basa Tupuy: la especificación de requisitos con casos de uso, los diagramas de clases de análisis y la estimación del esfuerzo con Cocomo II y Puntos de Función. Esto con el fin de que se pueda entender lo que propone Tupuy.

4.1.1 La especificación de requisitos con casos de uso

Los casos de uso fueron introducidos en 1987 como una herramienta de la técnica Objeto. Su utilización en los procesos de Ingeniería de Software fue propuesta por Ivar Jacobson y publicada en su libro “Object Oriented Software Engineering” [Jacobson, 1992]. Actualmente, su empleo se ha extendido aún más, debido a su inclusión en UML [OMG, 2008] por lo que su uso en el desarrollo de software orientado a objetos se ha vuelto altamente recomendable, por no decir obligatorio.

4.1.1.1 Actores

Para determinar la funcionalidad de un sistema a desarrollar, UML incluye dos elementos de modelado para este fin: el actor y el caso de uso. El actor representa una entidad externa que interactúa con el sistema. Las entidades externas podrían ser personas u otros sistemas. Es importante resaltar que los actores son abstracciones de papeles o roles y no necesariamente tienen una correspondencia directa con personas. La Figura 4-1 muestra la representación gráfica de los dos elementos anteriormente mencionados.



Figura 4-1. Representación gráfica de actor y caso de uso

Para encontrar actores del sistema, se puede buscar entre las categorías de personas, dispositivos de hardware, redes de computadoras u otro software. En un sistema de biblioteca, los actores podrían ser bibliotecario o cliente; este último si es que hay módulos de consulta de libros. Para el caso de un sistema de ventas, los actores podrían ser el cliente, si se realiza ventas por Internet, el vendedor y el sistema de facturación.

En un sistema, un usuario puede tomar el rol de como muchos actores; por ejemplo, en un banco, Juan Pérez podría ser cliente y operador dependiendo el momento y el uso que haga del sistema.

4.1.1.2 Los escenarios y los casos de uso

Un escenario, según Bruegge [BRUEGGE, 2002]: “Es una descripción concreta, enfocada e informal de una sola característica del sistema desde el punto de vista de un solo actor”; es decir, un escenario muestra la secuencia de pasos que se produce cuando un actor interactúa con el sistema en una situación específica y un tiempo determinado.

Un ejemplo de escenario para un sistema de biblioteca es el siguiente: “Juan Pérez se conecta al sistema de la Biblioteca Nacional a través de Internet. Juan Pérez selecciona realizar búsqueda y cuando aparece el formulario ingresa en título de libros la frase ‘especificación de requisitos’. El sistema encuentra un único libro y lo muestra, el libro de la biblioteca es ‘Especificación de Requisitos de Software’ de Alan Davis y código B 73-825” Otro ejemplo de escenario sería si Juan Perez hubiera ingresado como título ‘especificación’; otro, si hubiera ingresado ‘especificación de casos de uso’; etc.

Los escenarios se pueden clasificar en primarios y secundarios. Los escenarios primarios son aquellos a los que se denominan escenarios “happy day” porque están relacionados a aquellas situaciones en las que no existe ningún error o excepción. En cambio los escenarios secundarios están asociados a situaciones excepcionales o problemáticas. Los autores indican que los escenarios primarios conforman un 80% del total de escenarios y los secundarios, el 20% restante.

Cabe resaltar que no es necesario documentar los escenarios de manera formal. Esto quiere decir que carece de importancia la creación de documentos que describan todos los escenarios posibles del sistema, ya que su propósito es servir en la identificación de los casos de uso del sistema.

La diferencia entre escenarios y casos de uso radica en que un escenario es una instancia de un caso de uso [Bruegge, 2002]. El caso de uso es el que especifica todos los escenarios posibles para una parte de funcionalidad dada; es decir, todos los escenarios similares se agrupan en un solo caso de uso.

UML especifica que para representar gráficamente la relación entre un actor y caso de uso se debe trazar una línea que los una a la que se le denomina “relación de comunicación”. Por ejemplo, en el sistema de biblioteca las consultas de bibliografía por Internet por parte de Juan Pérez y cualquier otro cliente se puede agrupar en un solo caso de uso, al que se le puede denominar “Consultar bibliografía” (ver Figura 4-2).

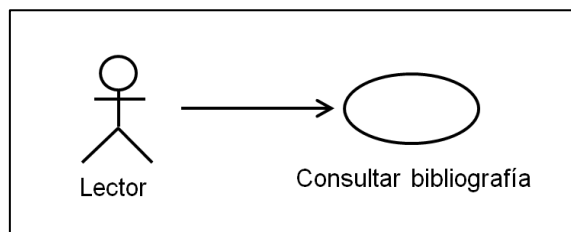


Figura 4-2. Diagrama de casos de uso para un sistema de biblioteca

4.1.1.3 La documentación de los casos de uso

Después de haber identificado los casos de uso, se tienen que indicar detalladamente la forma en la que el actor interactúa con el sistema. Esto se determina mediante la especificación y documentación de cada caso de uso.

Según RUP [IBM, 2006], la especificación de cada caso de uso debe contener lo siguiente:

1. **Precondiciones**, que detallan los estados en los que debe estar el sistema para que se pueda ejecutar el caso de uso; por ejemplo: para que un actor pueda ejecutar un caso de uso específico previamente ha

debido validarse en el sistema (ejecución de un escenario del caso de uso *login*).

2. **Flujo básico**, que muestra la secuencia de pasos que se va a producir en la mayoría de las veces en que ese caso de uso se ejecute. Es aquí donde están representados los escenarios primarios.
3. **Flujos alternativos**, que contienen las secuencias de pasos que se producirán como alternativas al flujo básico del caso de uso; es decir, especifican los pasos que se producirán en situaciones excepcionales. Es aquí donde están representados los escenarios secundarios.
4. **Postcondiciones**, que señalan el estado en que el sistema quedará luego de haberse ejecutado el caso de uso.

Algunos autores, por ejemplo Larman [Larman, 1999], reemplazan el título de “flujo básico” por “escenario primario” y el de “flujos alternativos” por el de “escenarios secundarios”. Esto podría generar cierta confusión entre el concepto de escenario y el de caso de uso, los cuales previamente se han explicado; es por ello que solo se considerará la propuesta de RUP.

4.1.1.4 Las relaciones entre caso de uso

UML [OMG, 2008] especifica que pueden emplearse tres tipos de relaciones entre casos de uso: *include*, *extend* y *generalization*. El empleo de estas relaciones, a sugerencia de Rosenberg [Rosenberg, 1999], debería ser opcional y no obligatoria; porque se pueden hacer buenas especificaciones de requisitos con casos de uso sin tener que emplear ninguna de esas relaciones.

La relación *include* se deberá determinar cuando la especificación de dos o más casos de uso contenga secuencias de acciones iguales. Para ello, la secuencia de pasos que se repite entre ellos, será extraída de esos casos de uso y se creará un nuevo caso de uso que los incluya. Por ejemplo, en la Figura 4-3, se muestra parte del diagrama de casos de uso de un sistema bancario. Para ese ejemplo, los casos de uso “Retirar efectivo”, “Depositar dinero” y “Transferir fondos” tienen una relación *include* con el caso de uso “Identificar cliente”; porque los escenarios de esos tres casos de uso necesitan ejecutar los pasos de “Identificar cliente”.

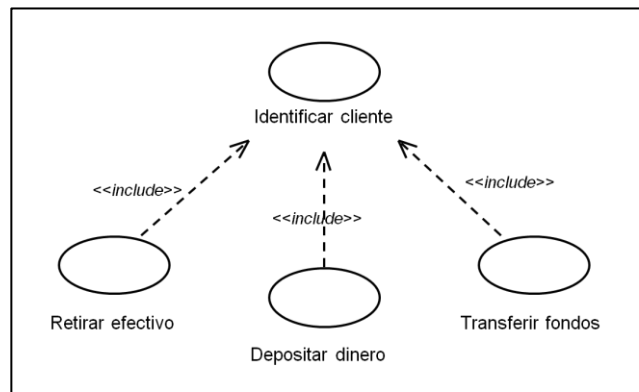


Figura 4-3. Ejemplo de diagrama de caso de uso para el sistema de un banco con *include*

La relación *extend* se produce cuando existe una secuencia de acciones que se producen en ocasiones excepcionales. Para ello, se creará un nuevo caso de uso que contenga dicha secuencia de pasos y dicho caso de uso extenderá la funcionalidad del caso de uso original. En el ejemplo mostrado en la Figura 4-4, la secuencia de pasos especificada en el caso de uso “Mostrar identificación de quien llama” solo se ejecutará de manera excepcional, para algún escenario del caso de uso “Realizar llamada”.

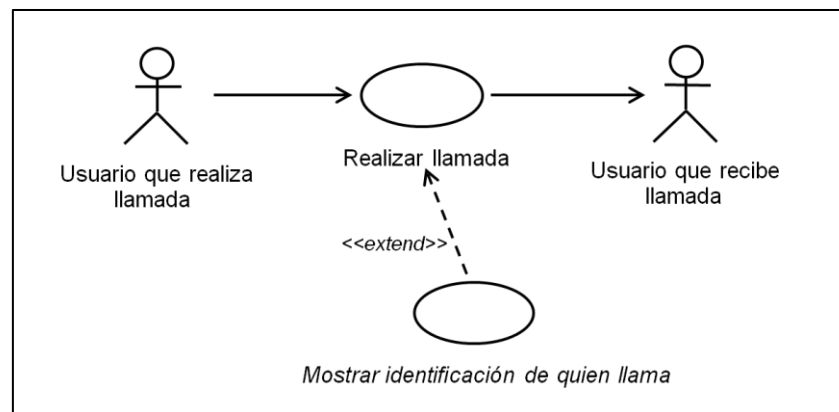


Figura 4-4. Ejemplo de diagrama de caso de uso con *extend*

En cuanto a la relación *generalization* se identifica cuando existen casos de uso cuyo propósito es similar y contienen secuencias de acciones parecidas. En ese caso, se crea un caso de uso genérico al cual se le denomina caso de uso padre, del cual heredan dos o más casos de uso. La relación *generalization* entre casos de uso es análoga a la relación de “herencia” entre clases. La Figura 4-5 muestra un ejemplo de casos de uso con relaciones de tipo *generalization*. En

ese ejemplo, el caso de uso “Colocar orden por Internet” se considera como un subtipo del caso de uso “Colocar orden”. Esto quiere decir que este último caso de uso “heredaría” los pasos de los flujos de “Colocar orden”, lo que en la práctica correspondería a ejecutar los pasos que se heredarían.

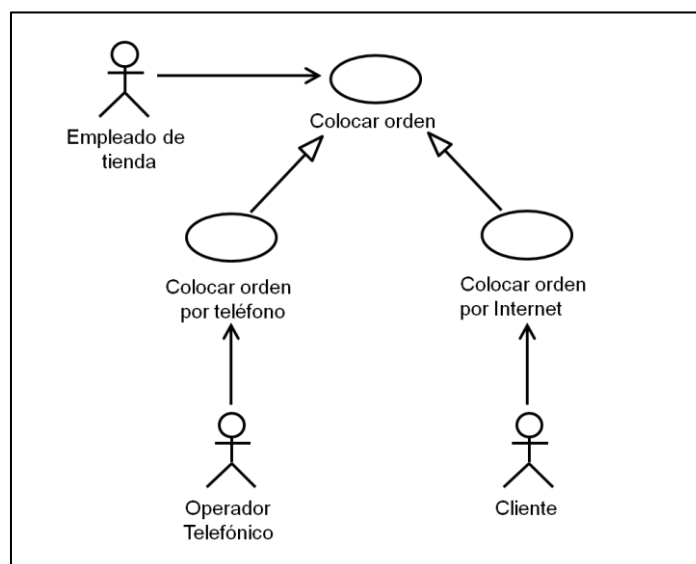


Figura 4-5. Diagrama de casos de uso con relación *generalization* entre casos de uso

4.1.1.5 Casos de uso concretos y casos de uso abstractos

Según la documentación de RUP [IBM, 2006] y de Moisiadis [Moisiadis, 2001] se pueden distinguir dos tipos de casos de uso: casos de uso concretos y casos de uso abstractos. Un caso de uso concreto es iniciado por un actor y constituye un flujo “**completo**” de eventos. Para este contexto, “**completo**” significa que una instancia del caso de uso, o escenario, realiza todos los pasos que son solicitados por un actor.

Un caso de uso abstracto puede ser incluido (relación *include*) por otro caso de uso, puede extender a otro caso de uso (relación *extend*) o puede generalizar (relación *generalization*) casos de uso. Cuando un caso de uso concreto es iniciado, una instancia de este caso de uso se crea. Esta instancia también exhibe el comportamiento especificado por sus casos de uso abstractos que tiene asociado. Esto quiere decir que no se puede crear instancias sólo a partir de casos de uso abstractos.

La distinción entre casos de uso es importante, porque los actores sólo pueden “**iniciar**” o “**ver**” los casos de uso concretos. Según las recomendaciones de RUP, los nombres de los casos de uso abstractos se deben colocar en letra cursiva. Por ejemplo, en la Figura 4-4 que muestra un ejemplo de *extend*, el caso

de uso “Mostrar identificación de quien llama” es un caso de uso abstracto; porque los pasos especificados en él solo pueden ser ejecutados mediante la ejecución de un escenario del caso de uso “Realizar llamada”.

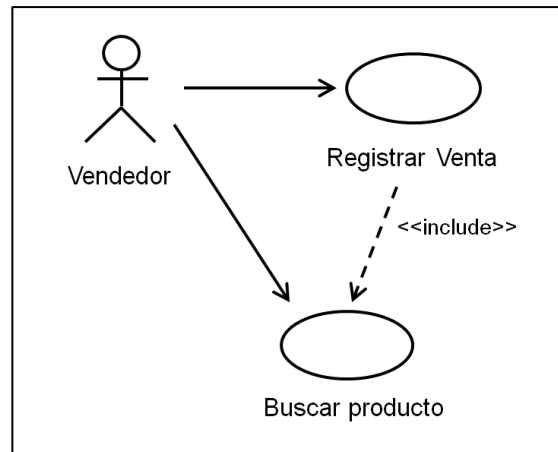


Figura 4-6. Ejemplo de caso de uso incluido no abstracto

Cabe resaltar que los casos de uso incluidos, extendidos y generalizados no necesariamente son abstractos. Por ejemplo, la Figura 4-6 muestra el caso de uso incluido “Buscar producto” que no es abstracto. En este ejemplo, se podría ejecutar un escenario de “Buscar producto”, si es que se le consulta a un vendedor sobre un determinado producto; o cuando al registrar una venta (caso de uso “Registra venta”), se tiene que buscar la información adicional del producto, si el cliente lo solicita.

4.1.2 El análisis con diagramas de clase

Según la propuesta de Jaaksi [Jaaksi, 1998], la fase de análisis se realiza después de la fase de captura de requerimientos y su propósito es entender el problema del dominio y del sistema a implementar. Como parte de esta fase, se realiza un diagrama de clases que documenta los conceptos del dominio del problema, Larman [Larman, 1999] le denomina modelo conceptual. En el caso del paradigma estructurado, se emplean diagramas Entidad-Relación [Chen, 1975]

Este modelo conceptual o diagrama de clases a nivel de análisis ofrece la ventaja de concentrarnos en los objetos del dominio y no en las entidades de software. Este tipo de diagrama muestra conceptos, asociaciones entre conceptos y atributos de conceptos. Es un error colocar clases correspondientes a diseño, como ventanas o bases de datos, en un modelo conceptual. Tampoco se deberían identificar métodos.

Según Kodaganallur [Kodanagallur, 2006], los patrones de análisis ayudan en la especificación de requisitos y tienen el potencial de mitigar los riesgos. Además, menciona que Silverston [Silverston, 2001] sugiere que aproximadamente el 50% de los requerimientos funcionales de un sistema pueden ser cubiertos con modelos genéricos aplicables a todas las organizaciones, que otro 25% puede ser modelado con modelos preexistentes a todas las organizaciones de un tipo de industria y que solo un 25% de los requisitos son específicos a la organización. Por ello, es importante el empleo de patrones.

Los patrones de análisis se ocupan en representar cómo los objetos y conceptos se relacionan en el mundo real, mientras los patrones de diseño se ocupan de una o más propiedades de un sistema software.

Muchos patrones de análisis han sido sugeridos; sin embargo, algunos de los más reconocidos son los propuestos por Martin Fowler [Fowler, 1997] y Peter Coad [Coad, 1997]. Coad, a diferencia de otros autores, incluye en sus patrones la relación de agregación entre clases (ver acápite 2.6.1.2 del capítulo 2 para mayor detalle de lo que es una relación de asociación, agregación y composición).

Los patrones de Coad a nivel de modelo conceptual se clasifican en patrones de transacciones, patrones de plan y patrones de agregación. La Figura 4-7 muestra un resumen de los patrones de Coad originalmente propuestos. Como se puede observar, los patrones de Coad no están representados en UML, pero Wohed [Wohed, 2001] muestra una representación de estos patrones en UML.

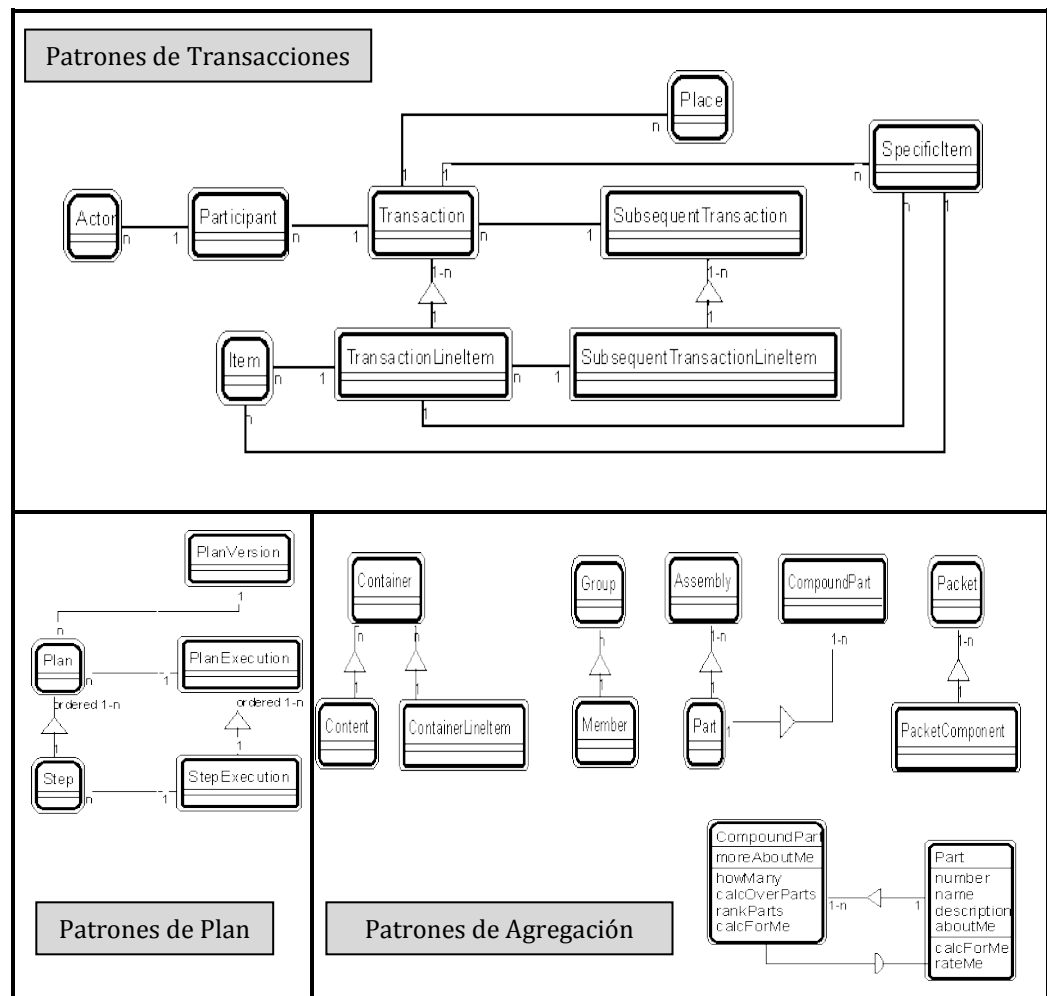


Figura 4-7. Resumen de los patrones de Coad

En la notación empleada por Coad (ver Figura 4-7), el triángulo que se incluye en algunas de las relaciones entre clases representa agregación y el semicírculo, herencia. Los patrones de Coad no consideran composición, porque este tipo de relación fue definido después de la publicación de los patrones. No obstante, al revisar las relaciones de agregación que muestran algunos de ellos, realmente deberían ser composiciones. Incluso, la representación que hace Wohed [Wohed, 2000] en UML de estos patrones no considera composición. Aunque la Figura 4-7 muestra un resumen de estos patrones, ellos originalmente fueron especificados empleando relaciones entre dos clases. La Figura 4-8 muestra los patrones de transacciones con ejemplos y como originalmente se indican en el libro de Coad [Coad, 1997], pero utilizando UML.

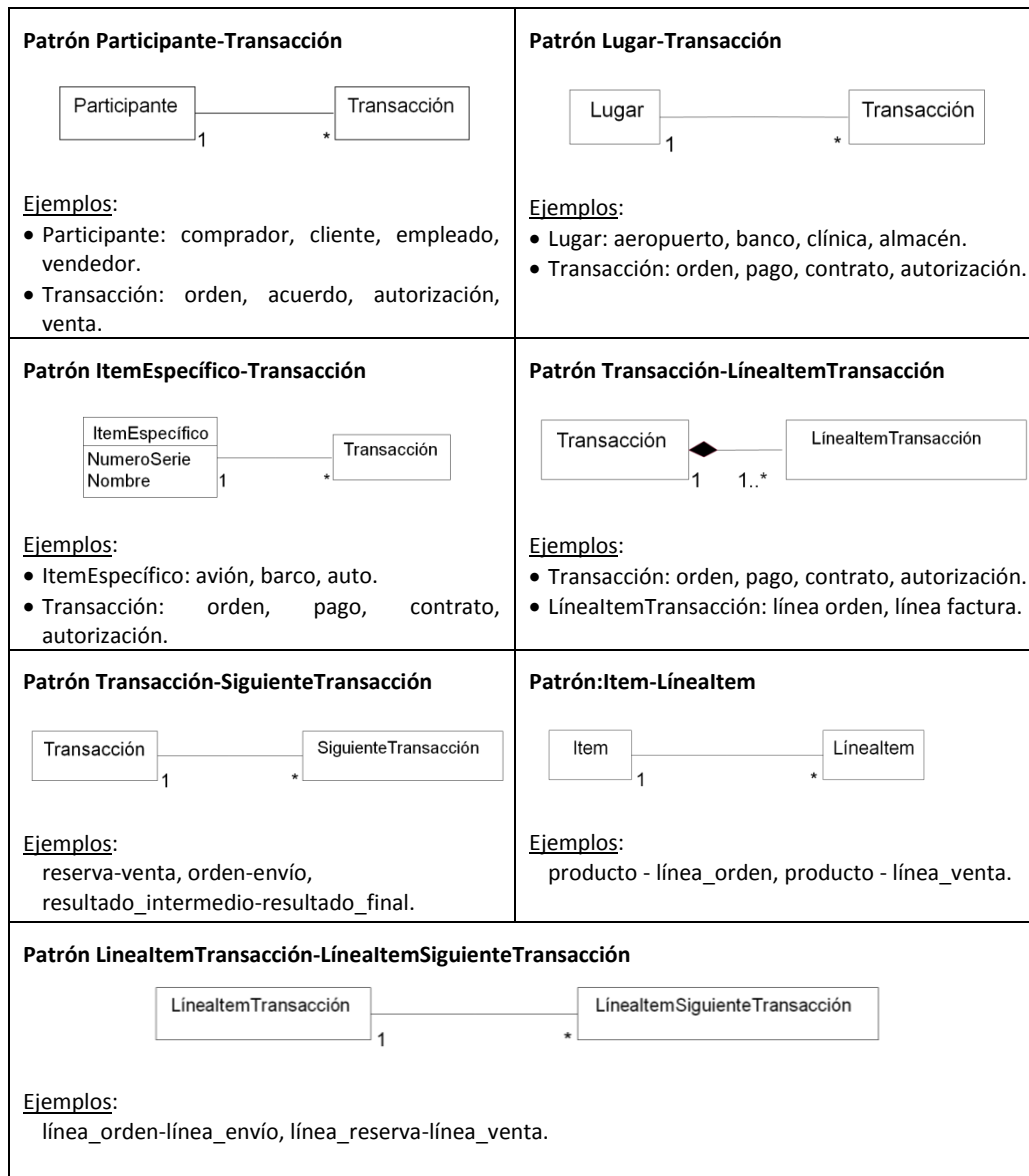


Figura 4-8. Patrones de transacciones de Coad en UML con ejemplos

Como se puede observar en la Figura 4-8, los patrones de Coad muestran las relaciones entre dos clases. Sin embargo, todos estos patrones de transacciones se pueden resumir tal y como se muestran en la Figura 4-9.

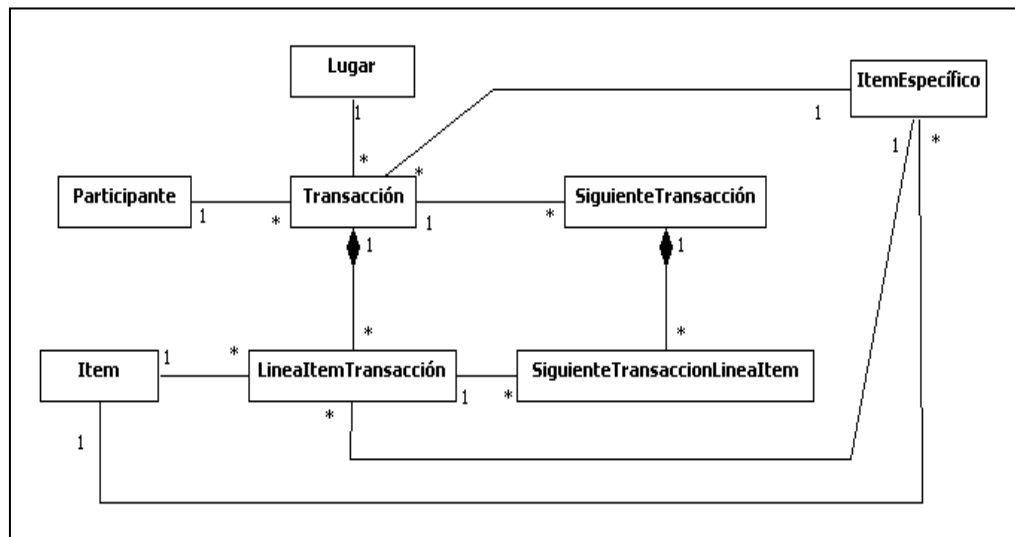


Figura 4-9. Patrones de plan de Coad en UML

El resumen de los patrones de plan se muestran en la Figura 4-10. Algunos ejemplos de plan son proyecto, plan de batalla, receta de cocina, plan, trabajo y procedimiento. En cuanto a los pasos, estos podrían ser tarea de plan, pasos tácticos, pasos de la receta de cocina, pasos de un trabajo y pasos de un procedimiento. En cuanto a EjecuciónPlan y EjecuciónPaso, estos especifican la información de las ejecuciones de un plan o de un paso como estado, fecha de inicio y fecha de fin de la ejecución.

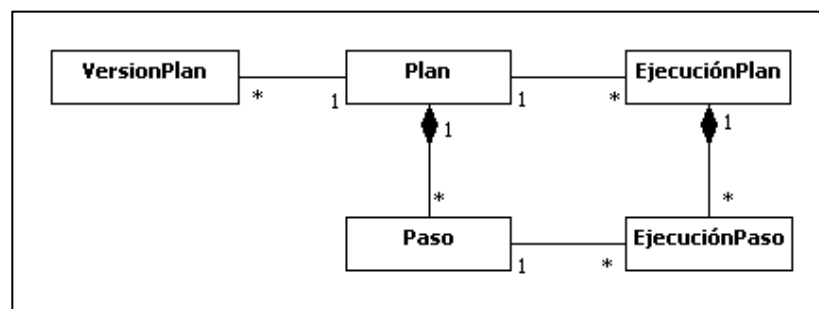


Figura 4-10. Resumen de patrones de plan de Coad en UML

Los patrones de agregación de Coad muestran conceptos que están relacionados mediante una relación todo/parte. La Figura 4-11 muestra este tipo de patrones. En algunos de ellos, se ha dejado como agregación, aunque según el dominio del problema, podría cambiarse a composición.

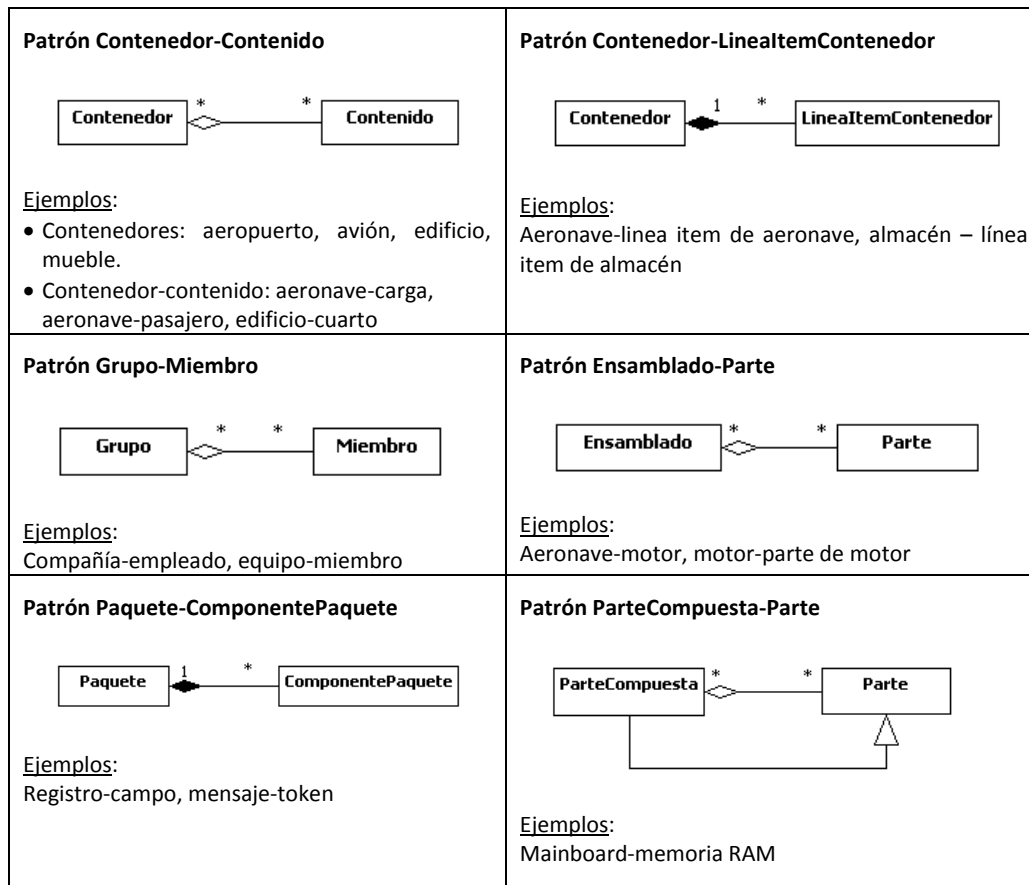


Figura 4-11. Patrones de agregación de Coad en UML

Como se puede observar en la Figura 4-11, solo se ha considerado composición en los patrones Paquete-ComponentePaquete y Contenedor-LineaItemContenedor. Este último es muy parecido a la forma del patrón Transacción-LineaItemTransacción; sin embargo, este último patrón está asociado al concepto abstracto de transacción (venta, pedido, etc) y el otro, a algo tangible como un almacén o una aeronave.

Los patrones originales de agregación indican una multiplicidad de 1 en el lado de la relación correspondiente al “todo” (lado que contiene el rombo); no obstante, se ha preferido dejarlo como multiplicidad de “muchos” porque en la mayoría de ejemplos dados correspondería a este tipo de multiplicidad. Sin embargo, esto debería cambiar según el dominio del problema.

4.1.3 Estimación del esfuerzo con Cocomo II y Puntos de Función

Cocomo, propuesto y desarrollado por Barry Boehm [Boehm, 1981], es uno de los modelos de estimación de costos mejor documentados y utilizados. El modelo permite determinar el esfuerzo y tiempo que se requiere en un proyecto de software a partir de una medida del tamaño del mismo expresada en el

número de líneas de código que se estimen para generar la creación del producto software. Debido a la complejidad de los proyectos de software, el modelo original fue modificado, denominándose al modelo actual Cocomo II [Boehm, 2000a]. La fórmula 4.1 pertenece a Cocomo y permite estimar el esfuerzo.

$$\text{Esfuerzo} = C1 \times EAF \times \text{Tamaño}^{P1} \quad (4.1)$$

En la fórmula 4.1, **Esfuerzo** es el esfuerzo calculado en meses-persona; **C1** es una constante de esfuerzo; **Tamaño** es el tamaño del producto software a desarrollar estimado en miles de líneas de código; **EAF** es el factor de ajuste de esfuerzo (*Effort Adjustment Factor* en inglés) que caracteriza el dominio, el personal, el entorno y las herramientas del proyecto de construcción de software; y **P1** es un exponente que caracteriza las economías de escala que fueron encontrados para proyectos de software de diferentes tamaños [Banker, 1994].

En cuanto a P1, hay autores como Fenton y Pfleeger [Fenton, 1997] que afirman que debería ser uno y otros autores como Hastings et. al [Hastings, 2001] que al utilizar información de proyectos de software, confirman que este exponente es cercano a uno y simplifican la fórmula eliminando este exponente.

Este modelo define, como parte de su fórmula de estimación, el factor de ajuste de esfuerzo, en inglés *effort adjustment factor* (EAF), que representa los efectos combinados de múltiples parámetros. Estos parámetros, conocidos en inglés como *Effort Multipliers* (EM), son empleados para capturar las características del desarrollo de software que afectan el esfuerzo para completar el proyecto. La Tabla 4-1 muestra los *Effort Multipliers*, multiplicadores de esfuerzo en castellano, correspondientes al modelo de post-arquitectura.

Tabla 4-1. *Effort Multipliers* para el modelo de Post-Arquitectura de Cocomo II

<i>Effort Multiplier</i>	Descripción
RELY	Fiabilidad requerida del software
DATA	Tamaño de la base de datos
CPLX	Complejidad del producto
RUSE	Reusabilidad requerida
DOCU	Documentación de acuerdo a las necesidades del ciclo de vida
TIME	Restricción de tiempo de restricción
STOR	Restricción de almacenamiento principal
PVOL	Volatilidad de la plataforma
ACAP	Capacidad de analistas
PCAP	Capacidad de programadores
PCON	Continuidad del personal
AEXP	Experiencia en aplicaciones

<i>Effort Multiplier</i>	Descripción
PEXP	Experiencia de plataforma
LTEX	Experiencia de lenguajes y herramientas
TOOL	Uso de herramientas de software
SITE	Desarrollo en múltiples lugares
SCED	Restricciones de calendario impuestas al equipo de desarrollo

Cada EM se califica como muy bajo, bajo, nominal (o promedio), alto o muy alto. Cada uno de estos tipos de calificación tiene asociado un peso que típicamente varía entre 0.5 y 1.5, siendo 1 el valor nominal o promedio. Para calcular el EAF, se deben multiplicar todos los EM. La fórmula 4.2 expresa lo indicado.

$$EAF = \prod_{i=1}^{17} EM_i \quad (4.2)$$

Cabe resaltar que para el modelo de diseño temprano se emplearía la fórmula 4.2 pero solo con 7 EM. La Figura 4-12 muestra la selección de los EM AEXP, experiencia en aplicaciones, y LTEX, experiencia en lenguajes y herramientas, como muy bajo (*very low*, VL, en inglés) en la herramienta de Cocomo II. Esta herramienta fue desarrollada en la Universidad del Sur de California². Como se puede observar en la Figura 4-12, el EAF calculado es 1.46 cuando todos los EM son nominales o promedio, a excepción de LTEX y AEXP.

² La herramienta se puede descargar en
http://sunset.usc.edu/csse/research/CocomoII/cocomo_main.html

base + Incr % = rating

Product: RELY DATA DOCU CPLX RUSE

base NOM NOM NOM NOM NOM

Incr% 0% 0% 0% 0% 0%

Platform: TIME STOR PVOL

base NOM NOM NOM

Incr% 0% 0% 0%

Personnel: ACAP AEXP PCAP PEXP LTEX PCOM

base NOM VLO NOM NOM VLO NOM

Incr% 0% 0% 0% 0% 0%

Project: TOOL SITE

base NOM NOM

Incr% 0% 0%

User: USR1 USR2

base NOM NOM

Incr% 0% 0%

EAF is also affected by Schedule

EAF: 1.46

OK Cancel Help

Figura 4-12. Pantalla del software de Cocomo II que calcula el EAF

En Cocomo II, se puede determinar el esfuerzo y tiempo de un proyecto de software a partir de los PFSA, lo cual supone una gran ventaja, dado que en la mayoría de los casos es difícil determinar el número de líneas de código de que constará un nuevo desarrollo, en especial cuando se tiene poca o ninguna experiencia previa en proyectos de software. Esto hace que ambos modelos – Puntos de función y Cocomo II– sean perfectamente compatibles y complementarios.

Según la técnica de PF, luego de determinar los PFSA, se debe evaluar el grado de influencia de catorce características del proyecto de software. Muchas de estas características son similares a los EM de Cocomo II (por ejemplo, funciones distribuidas, desempeño y reusabilidad). A cada característica se le asigna un peso que varía desde 0.0 a 0.05 y luego se suman todas ellas. A este resultado se le suma 0.65 para producir un factor de ajuste conocido como *Value Adjustment Factor (VAF)*. No obstante, existen críticas sobre si se debería emplear el VAF en la técnica de Puntos de función.

Sobre el VAF, Lokan y Abran [Lokan, 1999] mencionan que hay cuestionamientos entre muchos autores sobre si es apropiado darle a cada una de las características la misma escala de seis puntos (entre 0 y 5) y el mismo peso de 0.01 para poder calcularlo. Además, indica que Benyhaia et al [Benyhaia, 1990] hace otro cuestionamiento importante, en el que desde el punto de vista estadístico, los catorce factores de ajuste no deberían sumarse

porque se influenciarían mutuamente; por lo que deberían ser multiplicados y no sumados.

Cocomo II, como se puede observar en la fórmula 4.2, multiplica los EM, los cuales son similares a las características para determinar el VAF; por lo que estarían de acuerdo al cuestionamiento de Benyhaia et. al [Benyhaia, 1990]. Además, el VAF es incompatible con Cocomo II, porque el VAF indica que solo un máximo de 5% del esfuerzo estimado sería influenciado por las catorce características, pudiendo ser más. Por ello, Cocomo II utiliza solo PFSA para medir el tamaño del software y no el VAF, sino el EAF que es el resultado de la multiplicación de los EM.

4.2 Descripción de TUPUY

Tupuy está conformada por tres técnicas: UML₂FP, UCPD y Incremental-FP. La Figura 4-13 muestra las actividades que incluye Tupuy. Además, se indica con una nube, las actividades que se hacen con cada técnica.



Figura 4-13. Actividades de Tupuy y técnicas que incluye

Como se puede observar en la Figura 4-13, la actividad 1 se realiza con la técnica UML₂FPA, la actividad 2 con la técnica UCPD y las actividades 3 hasta la 6 pertenecen a la técnica Incremental-FP. A continuación, se explicará a detalle, lo que especifica cada una de las técnicas incluidas en Tupuy: UML₂FP (ver

Subsección 4.2.1), UCPD (ver Subsección 4.2.2) e Incremental-FP (ver Subsección 4.2.3).

4.2.1 UML₂PF

Las reglas para calcular PF empleando modelos orientados a objetos, a la que se ha denominado en conjunto como UML₂FP, se han definido teniendo en cuenta dos elementos de PF: cálculo de ficheros (acápite 4.2.1.1) y cálculo de transacciones (acápite 4.2.1.2).

En la revisión del estado de la cuestión (ver capítulo 2), se pudo determinar que las técnicas existentes no cumplen con lo que especifican los últimos manuales de PF [Ifpug, 2004] [Ifpug, 2009]. Por ello, para la definición de las reglas, se ha considerado como requisito que éstas cumplan con lo que definen los últimos manuales de la técnica.

4.2.1.1 Cálculo de ficheros

Según los resultados obtenidos en la revisión del estado de la cuestión (ver Capítulo 2), para definir las reglas para el cálculo de ficheros, se ha tomado en cuenta las relaciones que podrían existir entre las clases: asociación, agregación, composición, herencia y clase asociación. Las reglas definidas, al igual que las técnicas existentes, utilizan el diagrama de clases obtenido en la fase de análisis para determinar los PFSA relacionados a ficheros (ILF o EIF).

Como se explicó en el capítulo de introducción (ver Capítulo 1), los Sistemas de procesamiento de transacciones, *Transaction Processing Systems (TPS)* en inglés, son sistemas de información automatizados creados para procesar grandes cantidades de datos relacionados a transacciones rutinarias y son utilizados por un gran porcentaje de empleados de las empresas u organizaciones. En un estudio de Curtis et. al [Curtis, 1994], se probó la efectividad de la técnica de PF para medir los TPS.

Según Kodaganallur [Kodaganallur, 2006], los patrones de transacciones de Coad [Coad, 1997] (ver subsección 4.1.2 para mayor detalle) pueden ser utilizados para modelar la mayoría de transacciones de un sistema como ventas, pagos, órdenes, registro de estudiantes, etc. Por ello, y considerando que los ejemplos que se incluyen en los manuales de PF [Ifpug, 1999] [Ifpug, 2004] [Ifpug, 2009] están relacionados a TPS, se ha tomado en cuenta los patrones de Coad para definir las reglas de cálculo de ficheros. Esto quiere decir que las reglas deberían cubrir todos los casos que especifican estos patrones.

Para determinar los ficheros y cuántos DET y RET tienen, se deben seguir las siguientes reglas.

Regla 1: Composición

- a) Si ocurre lo mostrado en la Figura 4-14 y B no tiene una relación de asociación, agregación o composición con otra clase adicional a A, entonces

contar 1 fichero por A y B con 2 RET. Se contará un DET por cada atributo de cada clase.

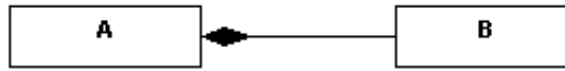


Figura 4-14. Diagrama de la regla 1a de ficheros

Esta regla cumple con todos los patrones de Coad que incluyen la composición en sus relaciones.

- b) Si ocurre lo mostrado en la Figura 4-15 y considerando que entre B y C la multiplicidad para B es de muchos y de C es de máximo de 1 (0..1 ó 1), entonces contar 1 fichero con 2 RET por A y B. Contar un fichero con 1 RET por C. Este conteo también se aplica si B tiene una relación de agregación con C. Se contará un DET por cada atributo de cada clase y para el caso del fichero de A-B se añadirá un DET más debido a la multiplicidad de muchos con C.

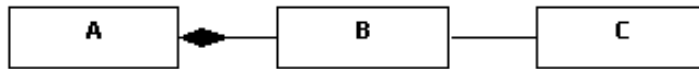
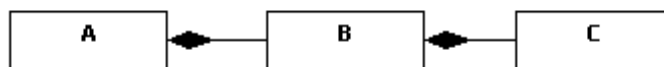


Figura 4-15. Diagrama de las reglas 1b y 1c de ficheros

Esta regla cumple con la mezcla de patrones Transacción-LinealItemTransacción y LinealItem-Item. También, cumple con la mezcla de patrones Contenedor-LinealItemContenedor y LinealItem-Item.

- c) Si ocurre como lo mostrado en la Figura 4-15 y considerando que entre B y C la multiplicidad para C es de muchos: contar 1 fichero con 2 RET por A y B. Emplear la regla 2 (regla siguiente) para determinar si C es otro fichero o un RET del fichero conformado por A y B.
- d) Si ocurre como lo mostrado en la Figura 4-16: contar 1 Fichero con 3 RET por A, B y C. En el caso que hubiera otra clase más que tenga una relación de composición con C, siendo C el todo, contar un RET más. Considerar esto último para composiciones sucesivas.

**Figura 4-16.** Diagrama de la regla 1d de ficheros**Regla 2: Asociación y Agregación**

En el caso que dos clases A y B estén relacionadas mediante asociación o agregación, seguir las reglas de la Tabla 4-1.

Tabla 4-2. Reglas para asociación y agregación de UML₂FP

Multiplicidad A	Multiplicidad B	Condición	La identificación sería
0..1	0..*	Son independientes	2 Ficheros y cada uno con un RET
1	1..*	Si B es independiente de A	2 Ficheros y cada uno con un RET
		Si B es dependiente de A	1 fichero, 2 RET
1	0..*	Si B es independiente de A	2 Ficheros y cada uno con un RET
		Si B es dependiente de A	1 fichero, 2 RET
0..1	1..*	Si A es independiente de B	2 Ficheros y cada uno con un RET
		Si A es dependiente de B	1 fichero, 2 RET
0..1	0..1	Son independientes	2 Ficheros y cada uno con un RET
1	1	Son dependientes	1 fichero y 1 RET
1	0..1	Si B es independiente de A	2 Ficheros y cada uno con un RET
		Si B es dependiente de A	1 fichero, 1 ó 2 RET
0..*	0..*	Son independientes	2 Ficheros y cada uno con un RET
1..*	1..*	Si B es independiente de A	2 Ficheros y cada uno con un RET
		Si B es dependiente de A	1 fichero, 2 RET
1..*	0..*	Si B es independiente de A	2 Ficheros y cada uno con un RET
		Si B es dependiente de A	1 fichero, 2 RET

Contar un DET por cada atributo de la clase. Adicionalmente, considerar un DET por la asociación/agregación en el lado que contenga la multiplicidad de muchos (*) en la relación, si es que es independiente (no considerarla si es dependiente).

Esta regla cumple con los patrones en la que las clases tienen las relaciones de asociación o agregación.

Regla 3: Herencia

En el caso de herencia se deberá considerar lo siguiente:

- Si las clases derivadas (clases hijas) tienen atributos que los diferencian entre sí, entonces contar un fichero y para ese fichero considerar un RET por cada clase hija. Contar como DET del fichero, los atributos de todas las clases que conforman la herencia.
- Si las clases derivadas tienen los mismos atributos, entonces contar un fichero con 1 RET. Contar como DET del fichero los atributos de todas las clases que conforman la herencia.

Esta regla no está asociada a ningún patrón de Coad, pero sí está relacionada a las reglas definidas por el Ifpug.

Regla 4: Clase Asociación

En el caso de clases asociación puede considerarse como un fichero independiente o como un RET de una o ambas clases, dependiendo las reglas del negocio.

Si la clase asociación es un fichero independiente, contar como DET los atributos de la clase y añadir dos más debido a las relaciones con las otras clases. Si la clase asociación es dependiente, añadir como DET del fichero que se crea a los atributos de la clase asociación y agregar un RET más a ese fichero.

Esta regla, al igual que la regla de herencia, no está asociada a ningún patrón de Coad, pero sí está relacionada a las reglas definidas por el Ifpug.

Regla 5: Consideraciones Adicionales

Si luego de haber empleado las reglas anteriores para identificar dos o más relaciones entre clases y aún existen clases que aún no se han identificado como ficheros, entonces se deberá realizar lo que se indica a continuación.

- a) Tomando como base lo mostrado en la Figura 4-17 y considerando que la relación entre A y B podría ser asociación, agregación, composición o herencia, y la relación entre A y C podría ser asociación, agregación o composición, pero no herencia; entonces:

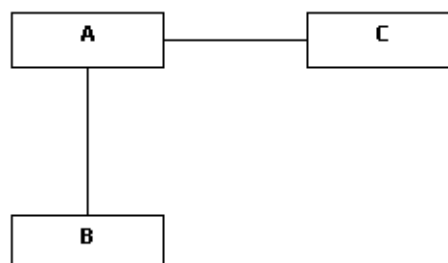


Figura 4-17. Diagrama de la regla 5 de ficheros

Si ya se determinó que A es un fichero independiente o dependiente de B, entonces seguir las reglas 1 y 2 para determinar si C es otro fichero o forma parte del fichero en el que se incluye A. En el caso que C sea un fichero independiente y la multiplicidad de A entre la relación A-C es de muchos, entonces añadir un DET adicional al fichero que pertenece A.

- b) Se aconseja que al inicio del cálculo determinar los ficheros debido a las relaciones de herencia y a las correspondiente a las clases asociación. Luego seguir las reglas anteriores, incluyendo a la 5a.

Finalmente, esta regla 5 deberá seguirse hasta que se hayan considerado todas las clases del diagrama.

Regla 6: Determinación de ILF o EIF y cálculo de PFSA

Tras definir los ficheros con sus RET y DET, se tiene que determinar si es ILF o EIF. Para ello, se siguen las especificaciones definidas por el Ifpug: se considera ILF si el fichero es mantenido por el sistema y EIF si solo es consultado y es mantenido por otro sistema. En este caso, se deberán revisar las especificaciones de casos de uso para determinar si el fichero es mantenido o no por el sistema.

Luego, se deben emplear lo especificado los manuales de PF [Ifpug, 2004] [Ifpug, 2009]; es decir que sabiendo qué fichero es ILF o EIF y conociendo sus DET y RET, se debe determinar la complejidad de cada uno de ellos. Para esta tarea, se debe emplear la Tabla 4-3.

Tabla 4-3. Tabla para determinar la complejidad de ILF o EIF

RET \ DET	1 a 19 DET	20 a 50 DET	51 o más DET
1 RET	Bajo	Bajo	Medio
2 a 5 RET	Bajo	Medio	Alto
6 a más RET	Medio	Alto	Alto

A continuación, y habiendo determinado la complejidad de cada fichero, se debe determinar el peso en PFSA de cada uno de ellos. Para ello, se debe utilizar la Tabla 4-4.

Tabla 4-4. Tabla para determinar el peso en PFSA de ILF o EIF

Complejidad de Fichero	PFSA para ILF	PFSA para EIF
Bajo	7	5
Medio	10	7
Alto	15	10

Regla 7: Notación gráfica para dependencia/independencia y para ficheros

Esta regla es opcional y solo se emplearía si es que se quiere mostrar gráficamente la dependencia entre dos clases. Para ello, según UML, la notación de dependencia entre dos elementos es una flecha punteada.

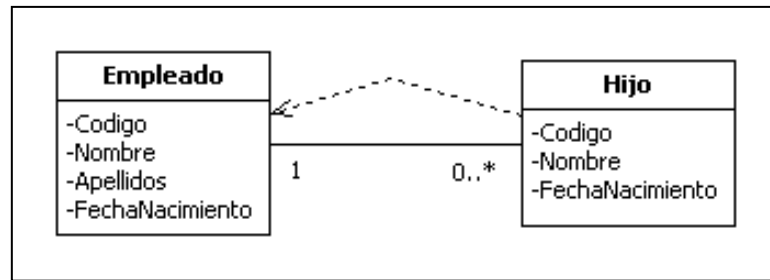


Figura 4-18. Ejemplo de la regla 7 con dependencias

La Figura 4-18 muestra un ejemplo tomado del manual de PF [Ifpug, 2009] en el que se incluyen las clases Empleado e Hijo. En este ejemplo, el hijo depende del Empleado; por lo tanto, adicionalmente a la asociación, se indica la relación de dependencia (flecha punteada). En el caso que la relación entre dos clases sea de composición, no es necesario indicar dependencia, ya que la composición representa una dependencia entre dos clases.

En cuanto a los ficheros, la representación más adecuada sería la de una clase a la que se le incluiría el estereotipo <<File>>, tal y como se muestra en la Figura 4-19.

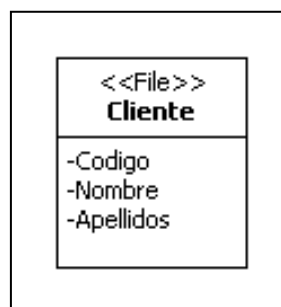


Figura 4-19. Ejemplo fichero de Puntos de función en UML

4.2.1.2 Cálculo de transacciones

La transacción de PF, originalmente denominada como *Transactional Function* por la técnica, es un proceso elemental que provee funcionalidad al usuario al procesar datos. Las transacciones desempeñan los procesos de guardar, actualizar, obtener y mostrar datos.

Para poder identificar las transacciones, se deben emplear las especificaciones de casos de uso, tal y como lo indica Fetcke [Fetcke, 1998]. Aunque existen algunas guías para especificar requisitos con casos de uso [Bittner, 2003] [Schneider, 2001] [Cockburn, 2000], no son obligatorias. Por ello, dos personas podrían representar de manera diferente el mismo sistema con casos de uso. Por ejemplo, la Figura 4-20 muestra dos diagramas: A y B, y ambos representan, a nivel de especificación de requisitos, lo mismo.

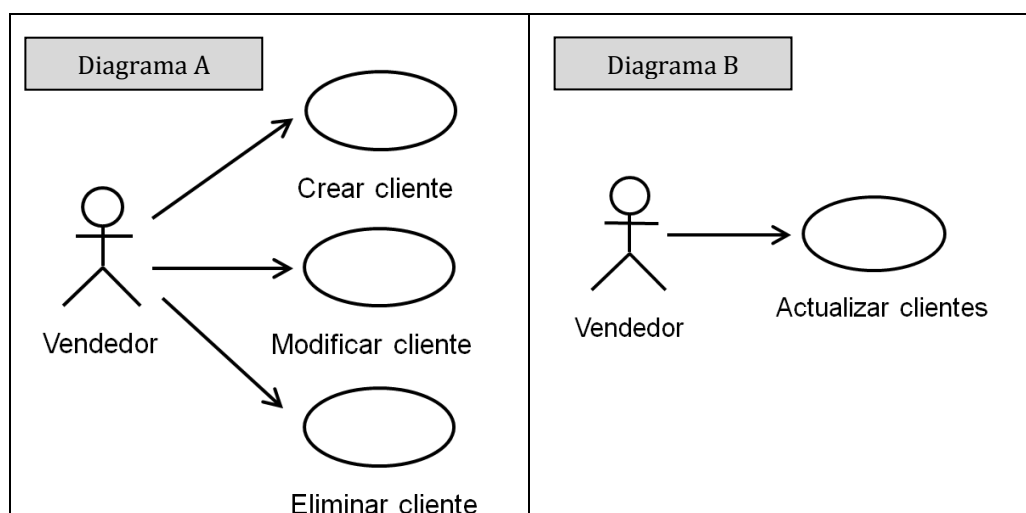


Figura 4-20. Ejemplos de diagrama de casos de uso para un mismo tipo de sistema de información

En la Figura 4-20, el diagrama A correspondería al estilo de identificación de casos de uso propuesto por Pender [Pender, 2003] y el diagrama B al propuesto por Bittner [Bittner, 2003]. Bittner, a diferencia de los ejemplos mostrados por Pender, sugiere que los casos de uso deben mostrar lo que el usuario necesita del sistema y no mostrar las funciones u opciones del menú que permitirán realizar lo solicitado; por ejemplo, según el ejemplo de la Figura 4-20, en un sistema donde se debe almacenar la información de los clientes, lo que al usuario le importa es actualizar la información de clientes. Esta actividad la podría realizar accediendo a las opciones del menú agregar, modificar y eliminar clientes; por lo tanto la funcionalidad del sistema será representada con el caso de uso “Actualizar cliente”.

Si se toma en cuenta las reglas de PF para identificar transacciones, cada caso de uso del Diagrama A en la Figura 4-20 correspondería a una transacción. Sin embargo, si se toma en cuenta el estilo sugerido por Bittner, el caso de uso Actualizar clientes tendría tres transacciones y estas transacciones no estarían siendo representadas cada una de ellas gráficamente.

A fin de considerar ambos estilos de casos de uso, se tuvo que seleccionar un elemento de UML que represente de manera más adecuada a las transacciones de PF; porque no se puede emplear los casos de uso para este fin. Adicionalmente, según la revisión del estado de la cuestión, se debería considerar un elemento que sea independiente de la metodología de desarrollo de software orientado a objetos considerada.

Las colaboraciones en UML son generalmente usadas para explicar como una colección de instancias cooperan entre sí para lograr una tarea o conjuntos de tareas [OMG, 2008]. Una colaboración muestra la implementación o realización (*realization* en inglés) de un caso de uso en términos de relaciones entre clases y objetos y su interacción para lograr una funcionalidad deseada [Eriksson, 2004]. Para explicar una colaboración, se requiere un número de diagramas que muestra el contexto y la interacción entre los elementos que colaboran como diagramas de secuencias, comunicación, actividades, etc. La colaboración es representada mediante una elipse con líneas punteadas.

RUP [IBM, 2006] propone el concepto de *Use Case Realizations*, realización o implementación de casos de uso en castellano. Según RUP, una “realización de casos de uso” representa la perspectiva de diseño de un caso de uso (CU). La razón de separar las realizaciones de CU de su caso de uso es para permitir que los casos de uso se puedan manejar de manera independiente de sus realizaciones.

Al revisar el concepto de transacción de PF, se puede notar que estos están asociados a una funcionalidad específica que se va a diseñar y luego programar. Sin embargo, no se puede especificar un caso de uso por transacción, porque no hay coincidencia con lo que mencionan los autores para identificar casos de uso. Por lo tanto una solución intermedia sería representar cada transacción como una colaboración. La Figura 4-21 muestra los estilos de casos de uso que se mostraron para un mismo sistema en la Figura 4-20 y las transacciones de PF asociadas a cada una de ellos. Las transacciones son representadas como colaboraciones (elipses con líneas punteadas) y la flecha representa realización.

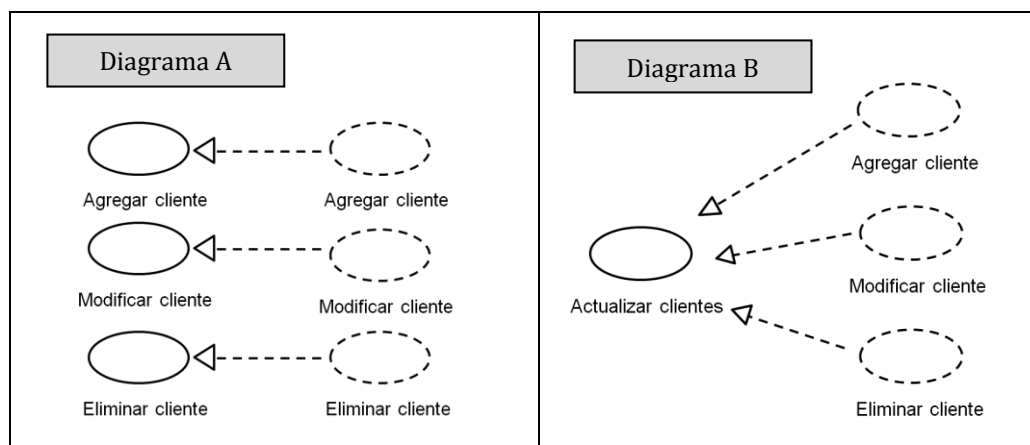


Figura 4-21. Casos de uso y transacciones asociadas

Como se puede observar en la Figura 4-21, si bien la cantidad de casos de uso es diferente al emplear diferentes estilos de identificación, las transacciones identificadas son las mismas para ambos estilos. La ventaja de hacer esto es que se mantenga separada la especificación de casos de uso del cálculo de PF. Además, esto quiere decir que sin importar el estilo de identificación de casos de uso, el conteo de PF será el mismo. Otra ventaja es que se han identificado, previamente al diseño, las funcionalidades que se tienen que implementar; es decir, que el desarrollador tendría que realizar los diagramas necesarios a nivel de diseño por cada transacción para que esto pueda ser finalmente programado.

Aunque *Use Case Realization* fue inicialmente una propuesta de RUP, el concepto de colaboración es parte de UML, por lo que no depende de una metodología de desarrollo específica. Además, OMG Press, entidad que se encarga de administrar UML, ha publicado libros como [Eriksson, 2004] que explican el concepto de realización de casos de uso. Por lo que se podría decir que esto ya no formaría parte de una metodología específica.

Tras identificar las transacciones, se seguirá con lo que indican los manuales de PF para calcular el peso de cada una de las transacciones. Para ello, se tomará como base la parte de las especificaciones de caso de uso correspondiente a cada transacción o los prototipos de pantallas asociados a ellas. Este cálculo puede hacerse más simple con la ayuda de una herramienta integrada a un modelador de UML. Por ejemplo, Tupux³ [Balbin, 2009] es una herramienta desarrollada para cumplir este propósito y una pantalla de ella se muestra en la Figura 4-22.

³ Tupux fue desarrollado por tres egresados del programa de pregrado de Ingeniería Informática de la PUCP. Esta herramienta automatiza Tupuy.

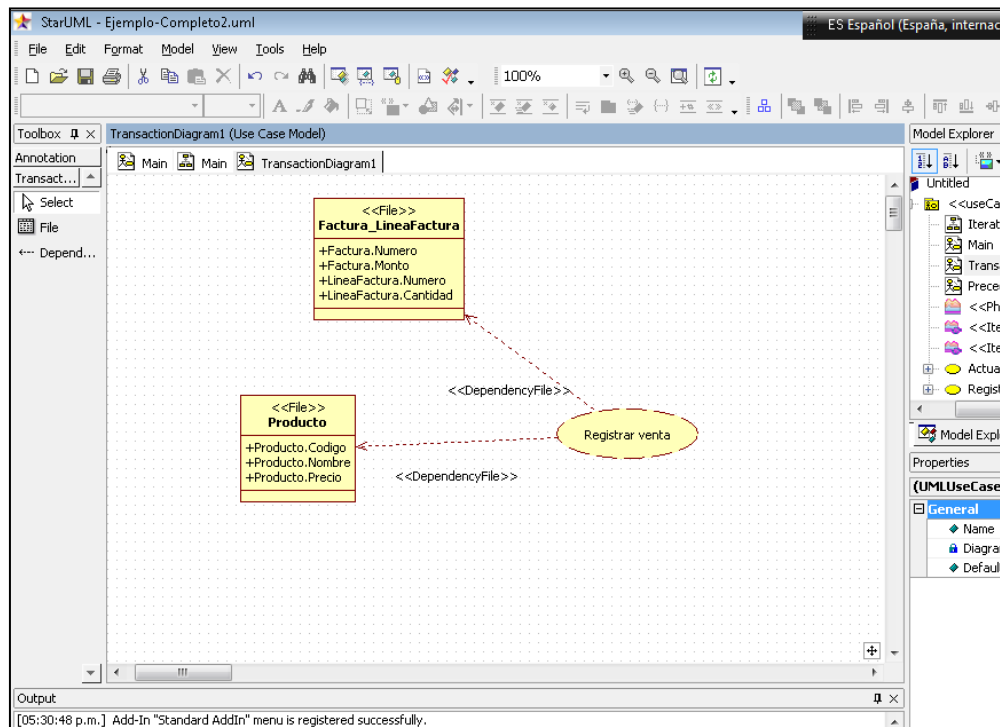


Figura 4-22. Pantalla de StarUML y Tupux en el que se muestra la transacción Registrar venta y sus ficheros asociados.

Tupux es un plug-in de StarUML que permite calcular los PF tomando como base diagramas orientados a objetos. En la Figura 4-22, se muestra una pantalla de StarUML con Tupux en el se representa a los ficheros *Producto* y *Factura_LineaFactura* como estereotipos de clases, y la transacción *Registrar venta* como una colaboración. Gráficamente, la relación entre los ficheros y la transacción se realiza mediante una dependencia “DependencyFile”⁴.

Al dar clic derecho sobre la transacción, se muestra una ventana que presenta todos los DET de los ficheros relacionados a las transacciones (ver Figura 4-23). En esta ventana se seleccionan los DET de la transacción, se añaden más DET en *Additional DET*, se indica si hay DET debido a capacidad de mensaje en *Send Message* o por habilidad para especificar una acción en *Generate Action*. También, se podrá seleccionar el tipo de transacción: EI, EO o EQ.

⁴ La definición del estereotipo “DependencyFile” no forma parte de la propuesta de esta tesis y corresponde a uno de los desarrolladores de la herramienta.

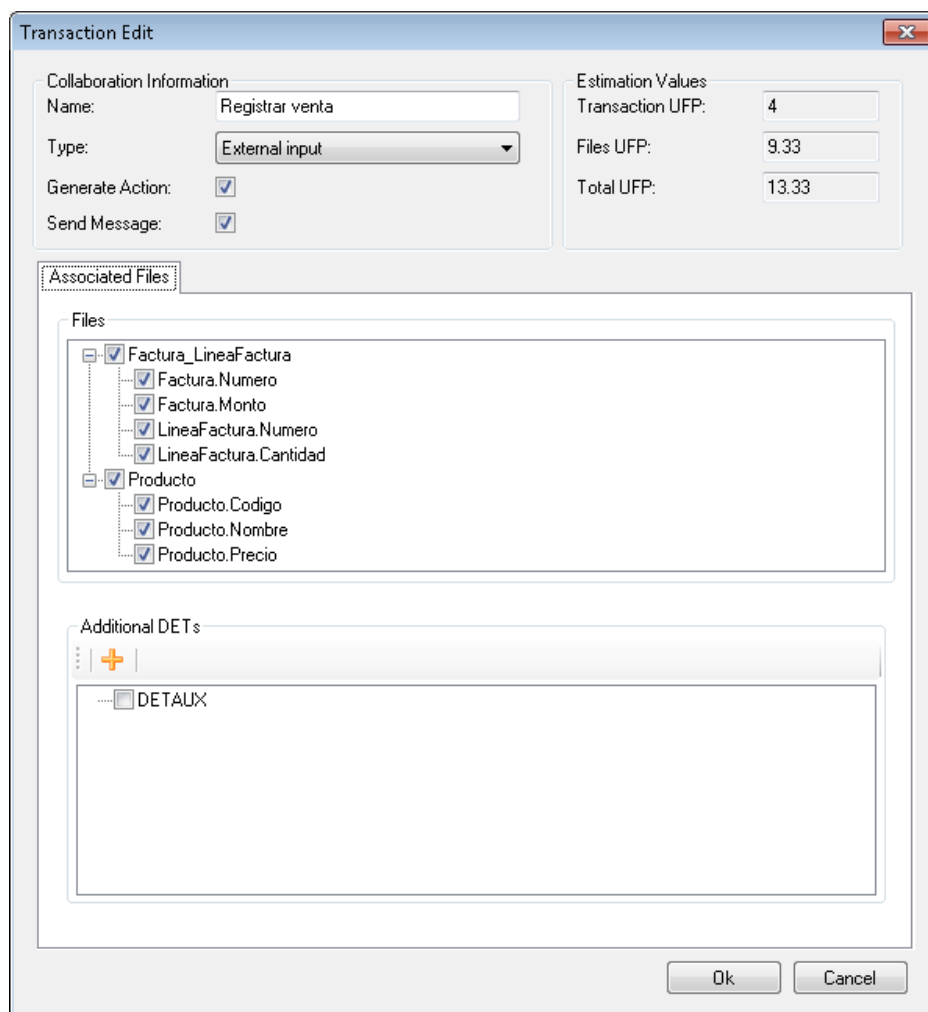


Figura 4-23. Pantalla para seleccionar DET de la transacción Registrar venta.

Ya no sería necesario indicar cuántos FTR tiene una transacción, porque las relaciones de *DependencyFile* entre ficheros y transacciones ya estaría mostrando la dependencia entre ellos. Por ello, Tupux se encarga de determinar cuántos FTR tiene cada transacción.

Como se puede notar, el cálculo de PF se puede integrar a una herramienta que permita hacer diagramas de UML y facilitar de esta manera el proceso de cálculo. En este caso, ya no sería necesario crear otros tipos de diagramas de UML, como los de secuencia o comunicación, para determinar los PFSA para las transacciones.

Cabe resaltar que Tupux genera los ficheros automáticamente según las reglas indicadas en el acápite 4.2.1.1, acápite anterior. Para ello, si dos clases son dependientes, se tiene que mostrar mediante una relación de dependencia de UML para que Tupux pueda generarlos como un solo fichero, tal y como se indica en la regla 7 para el cálculo de ficheros (acápite 4.2.1.1).

4.2.2 Use Case Precedence Diagram

Aunque existen muchas técnicas para determinar la secuencia de construcción de requisitos siguiendo esquemas de priorización, las cuales fueron incluidas en el capítulo de estado de la cuestión (ver Capítulo 2), la mayoría de ellas no considera la perspectiva o punto de vista del desarrollador en términos de facilidad de construcción para definir esas prioridades.

Como una práctica establecida, incluso comentada en foros de Internet⁵, se conoce que las tablas de una base de datos relacional de un sistema de información se pueden clasificar en los siguientes tipos:

- **Master Table**, tabla maestra en castellano, si contiene datos cambian con poca frecuencia; por ejemplo: información de los clientes de una empresa.
- **Transaction Table**, tabla de transacciones en castellano, si contiene datos que son modificados de manera frecuente; por ejemplo: órdenes de pedido de los clientes.

Cabe resaltar que muchos autores como Senn [Senn, 1992] dan una definición diferente a “tablas maestras” y “tablas de transacciones” que no son las mismas a las empleadas por muchos profesionales en la práctica.

Además de obtener la opinión de algunos profesionales con experiencia en proyectos de software, se aplicó una encuesta entre estudiantes de pregrado de la Pontificia Universidad Católica del Perú para determinar la secuencia que provee mayor facilidad para construir un sistema de información desde el punto de vista del desarrollador, teniendo en cuenta que las funcionalidades del software están relacionadas al concepto de tablas maestras y tablas de transacciones. Los resultados de la encuesta y de la opinión de los profesionales, considerando también como funcionalidad de un sistema de información a los reportes, muestran que la secuencia de construcción más adecuada, según la facilidad de construcción para el desarrollador, es la siguiente:

1. Mantenimiento de tablas maestras
2. Mantenimiento de tablas de transacciones
3. Reportes

Según esta clasificación, se pueden identificar tres tipos de casos de uso: casos de uso que se encargan de mantener tablas maestras, casos de uso que se encargan de mantener tablas de transacciones y casos de uso que se encargan de mostrar y generar reportes

Tomando en cuenta la clasificación de las tablas de una base de datos relacional y los casos de uso asociados a su actualización, se propone el empleo

⁵ Un ejemplo de un foro que trata el tema, se puede ver en <https://forums.sdn.sap.com/thread.jspa?messageID=4838177> (consultado en noviembre de 2011)

del diagrama denominado *Use Case Precedence Diagram* o UCPD, (diagrama de precedencias de casos de uso, en castellano). Este diagrama, adicionalmente a las relaciones entre casos de uso que especifica UML (*include*, *extend* y *generalization*), propone la inclusión de una nueva relación: **antecede**. La idea de este diagrama fue tomado de Doug Rosenberg [Rosenberg, 1999] quien propone el uso de un diagrama similar y especifica las relaciones *precede* e *invoke* entre casos de uso.

Según el estudio del estado de la cuestión, la relación *precede* entre casos de uso también es mostrada por Van Den Berg [Van Den Berg, 1999]. Aunque inicialmente se empleó para UCPD la relación *precede*, se prefirió el empleo de *antecede* para que no se confunda con las propuestas de Rosenberg y de Van Den Berg. La Figura 4-24 muestra un ejemplo de diagrama de precedencias.

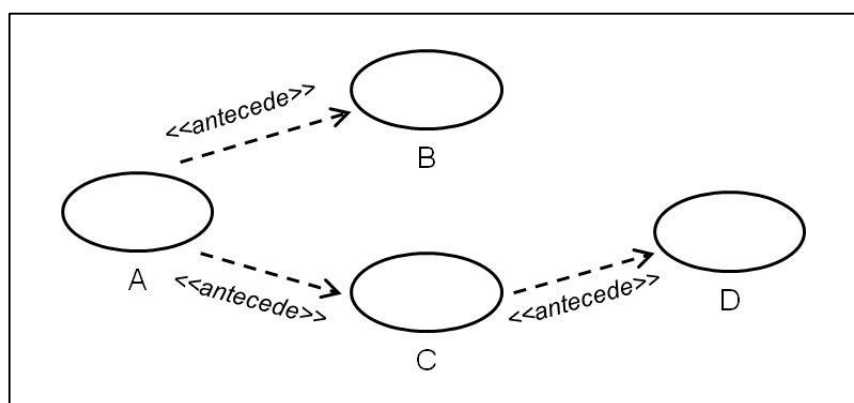


Figura 4-24. Ejemplo de UCPD

Para obtener las relaciones de precedencia entre casos de uso, se deben seguir las reglas que se muestran a continuación.

Regla 1:

Un caso de uso U1 precede otro caso de uso U2 si hay una precondition que corresponde a la ejecución de un escenario en U1 que debe ser completado antes de ejecutar un escenario U2. Por ejemplo, para ejecutar un escenario del caso de uso “Hacer reserva” para un sistema de ventas, el actor debe haber sido validado en el sistema, es decir se ha debido ejecutar el caso de uso login o “Validar usuario”. Por lo tanto este caso de uso precede al caso de uso “Registrar reserva”. El diagrama que muestra la precedencia de este ejemplo se presenta en la Figura 4-25.

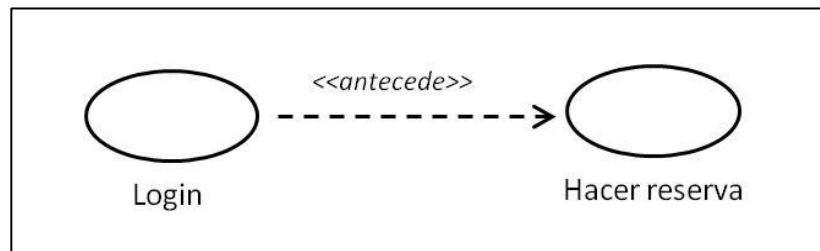


Figura 4-25. Ejemplo de la Regla 1 de UCPD

Regla 2:

Un caso de uso U1 precede a otro caso de uso U2 si U1 necesita información que es registrada por U2. Por ejemplo, para realizar el pago de una reserva, ésta ha debido ser ingresada previamente y es por ello que el caso de uso “Hacer reserva” precede al caso de uso “Pagar reserva”. La Figura 4-26 muestra el diagrama con la precedencia de estos casos de uso para el ejemplo dado.

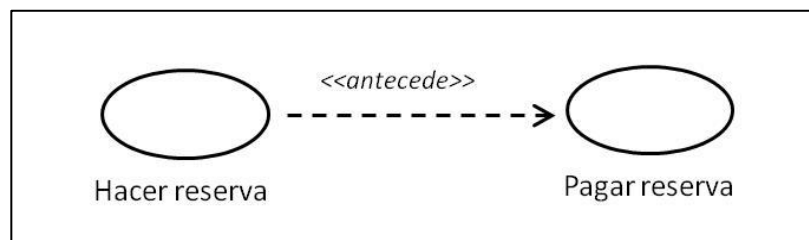


Figura 4-26. Ejemplo de la Regla 2 de UCPD

Cabe resaltar que en un UCPD solo se consideran los casos de uso que son concretos y no se incluyen los casos de uso abstractos. Según lo explicado en la subsección 4.1.1.5, algunos casos de uso incluidos, extendidos y generalizados pueden ser abstractos.

Al realizar un diagrama de precedencias de casos de uso, podría producirse casos como el mostrado en la Figura 4-27, en el que el caso de uso C tiene una precedencia con el caso de uso A y el caso de uso B; sin embargo, el caso de uso B también tiene como precedencia al caso de uso A.

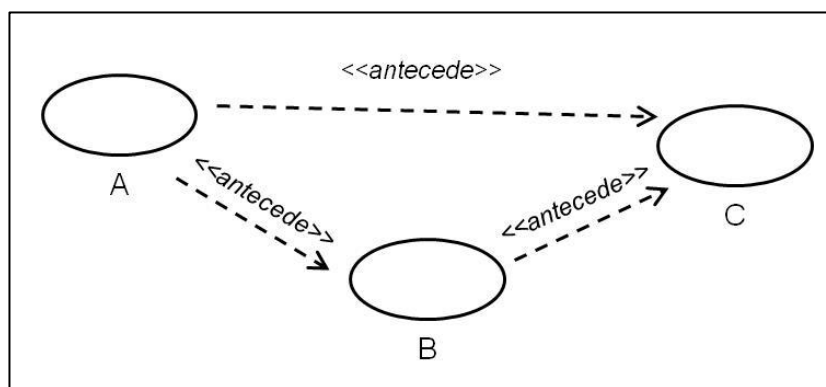


Figura 4-27. Ejemplo de diagrama de UCPD

Para estas situaciones, se podría simplificar el diagrama al eliminar la relación de precedencia entre los casos de uso A y C, porque el caso de uso B ya tiene como precedente al caso de uso A. El diagrama simplificado se muestra en la Figura 4-27. Cabe resaltar que ambos diagramas son válidos, no obstante el diagrama simplificado podría ser más claro de entender.

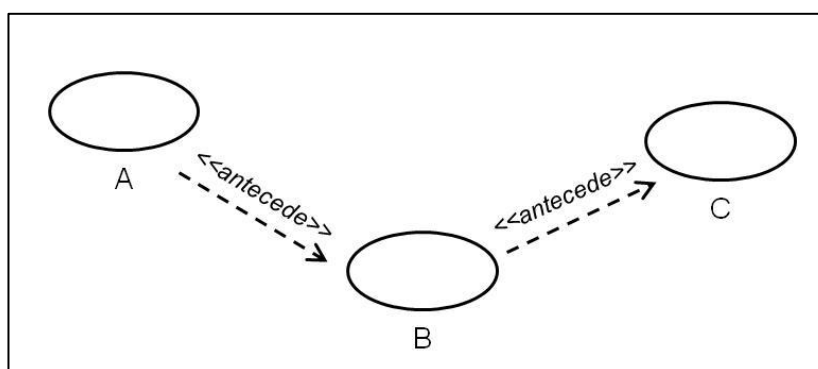


Figura 4-28. Diagrama simplificado del ejemplo de la Figura 4-27

Hay que tener en cuenta, siguiendo el ejemplo de la Figura 4-27, que podría ocurrir que hayan más casos de uso con relación de precedencia entre A y C, pues en esos casos, también se aplicaría la simplificación que se ha explicado.

4.2.3 Incremental-FP

La siguiente técnica a emplear, después de determinar la secuencia de construcción de casos de uso y de realizar el cálculo de PFSA empleando UML₂FP, es Incremental-FP, la cual permite determinar el esfuerzo que tomará

la construcción de cada incremento. La Figura 4-29 muestra las actividades que propone esta técnica.

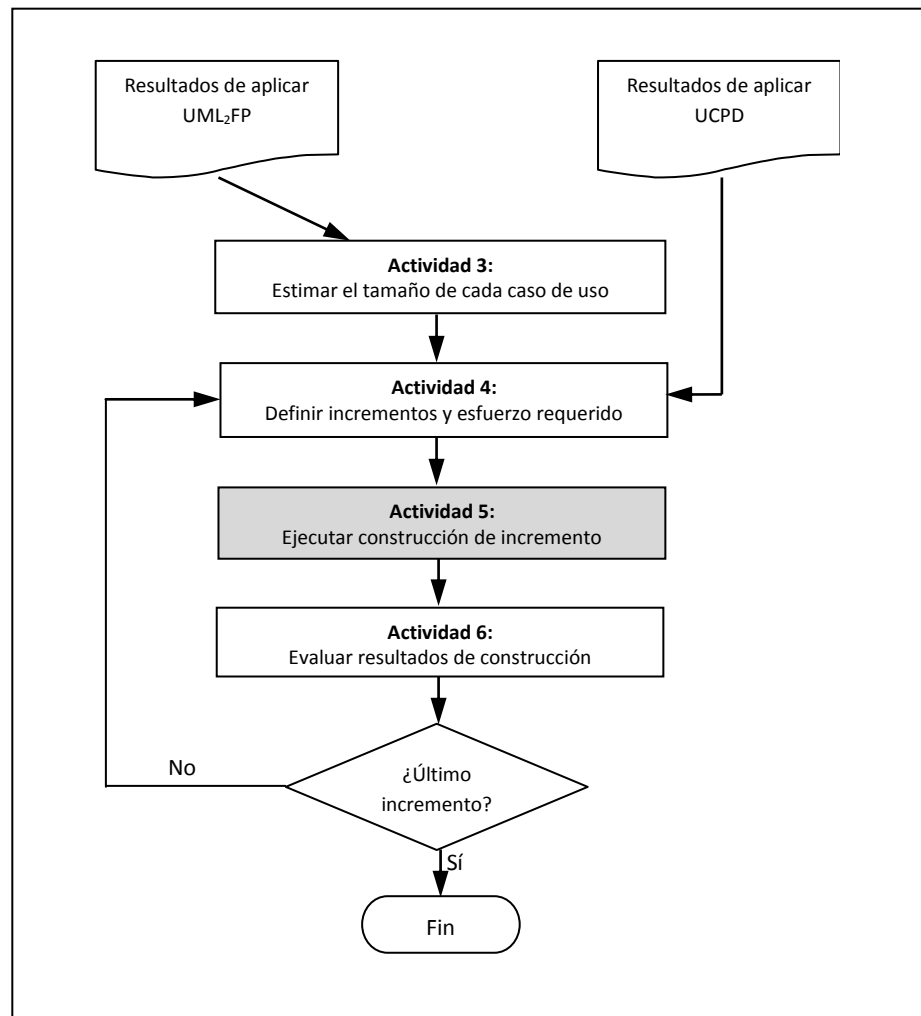


Figura 4-29. Actividades de Incremental-FP

La actividad 5, que se muestra en la Figura 4-29 en gris, no forma parte de la técnica, porque corresponde al desarrollo en sí del incremento. No obstante, se ha incluido en el diagrama para que se pueda entender la secuencia de ejecución de las actividades.

4.2.3.1 Estimar el tamaño de cada caso de uso (actividad 3)

La primera actividad, actividad 3 en la Figura 4-29, consiste en determinar los PFSA para cada iteración. El criterio seguido para adaptar la técnica de PF para ciclos de vida iterativos-incrementales consiste en considerar que el resultado del cálculo de los puntos de función sin ajustar para todo el proyecto

(PFSA_Total), deberá ser igual a la suma de los PFSA calculados para cada incremento por separado (PFSA(i) = PFSA del incremento “i”). La representación de lo indicado se muestra en la fórmula 4.3 que se muestra a continuación.

$$PFSA_Total = \sum_{i=1}^n PFSA(i) \quad (4.3)$$

Uno de los aportes de esta técnica consiste en determinar los PFSA correspondientes a los ficheros lógicos internos (ILF) y ficheros de interfaz externa (EIF) para cada caso uso. Inicialmente, se calculan cuántos PFSA debido a los ficheros tiene cada transacción. Para ello, esta técnica propone utilizar la fórmula 4.4., en la que los PFSA de un fichero específico (ILF o EIF) son repartidos proporcionalmente entre todas las transacciones encontradas en un software.

$$PFSA_FichTr(j) = \sum_{i=1}^n \left[\frac{1}{TTrPF(i)} \times PFSA_Fich(i) \right] \quad (4.4)$$

En la fórmula 4.4, **PFSA_FichTr(j)** son los PFSA para una transacción “j” debido a los ficheros, **TTrPF(i)** es el número total de transacciones que utiliza un fichero “i”, **PFSA_Fich(i)** son los PFSA del fichero “i”, “i” es el ILF/ EIF utilizado en la transacción “j” y “j” es una transacción.

Luego, se calculan los PFSA de cada caso de uso. Para ello, se emplean los PFSA correspondientes a ficheros de cada transacción asociada a cada caso de uso y que se han obtenido con la fórmula 4.4. A estos, se le añaden los PFSA debido a las transacciones. La fórmula que se muestra a continuación, fórmula 4.5, muestra lo indicado.

$$PFSA_CU(k) = \sum_{j=1}^n [PFSA_Tr(j) + PFSA_FichTr(j)] \quad (4.5)$$

En la fórmula 4.5, **PFSA_CU(k)** son los PFSA para un caso de uso “k”, **PFSA_Tr(j)** es la cantidad de PFSA de la transacción “j”, **PFSA_FichTr(j)** es la cantidad de PFSA de los ficheros correspondiente a la transacción “j” y “j” es la transacción correspondiente al caso de uso de uso “k”.

4.2.3.2 Definir incrementos y esfuerzo requerido (actividad 4)

La segunda actividad, actividad 4 en la Figura 4-29, consiste en definir los incrementos y el esfuerzo requerido. Para ello, se emplea el diagrama UCPD para seleccionar qué caso de uso se va a construir en cada incremento. Luego, para poder calcular el esfuerzo estimado, primero se debe calcular la productividad estimada para el primer incremento. Esto se calcula teniendo como base la productividad histórica de proyectos previos, la cual se mide en horas/PFSA, y el EAF histórico. También, se debe emplear el EAF de Cocomo II estimado para la primer incremento. Todo esto está representado en la fórmula 4.6. Como se explicó en la Sección 4.1.3, sería mejor utilizar el EAF de Cocomo II que el VAF de Puntos de función, ya que este último no sería muy adecuado desde el punto de vista estadístico [Benyhaia, 1990].

$$ProductividadEstimada_Incr1 = \frac{EAF_Incr1 \times ProductividadHistorica}{EAF_Historico} \quad (4.6)$$

En la formula 4.6, **ProductividadEstimada_Incr1** corresponde a la productividad estimada para el primer incremento, medida en horas/PFSA; **EAF_Incr1** es el EAF de Cocomo calculado para el primer incremento; **ProductividadHistorica** es el esfuerzo obtenido en proyectos previos; y **EAF_Historico** es el EAF que está relacionado al esfuerzo histórico.

La fórmula 4.6 que se emplearía para el primer incremento se puede generalizar para los demás incrementos. La fórmula 4.7 muestra cómo calcular la productividad estimada a partir del segundo incremento, tomando en cuenta la productividad real obtenida en incrementos previos.

$$ProductividadEstimada_Incr(i) = \frac{EAF(i) \times ProductividadReal_Incr(i-1)}{EAF(i-1)} \quad (4.7)$$

En la formula 4.7, **ProductividadEstimada_Incr** corresponde a la productividad estimada para el incremento "i"; **EAF** es el EAF de Cocomo calculado para el incremento "i" o incremento "i-1"; **ProductividadReal_Incr** es el esfuerzo real obtenido en el incremento previo, es decir en el incremento "i-1"; e "i" corresponde al número de incremento.

Como se puede observar en la fórmula 4.7, para estimar la productividad de un nuevo incremento se tomaría la productividad real del incremento previo. Sin embargo, esto podría funcionar si fueran pocos incrementos, pero lo más adecuado sería emplear el promedio de las productividades reales obtenidas en incrementos previos. La fórmula 4.8 muestra la mejora de la fórmula 4.8.

$$ProductividadEstimada_Incr(i) = EAF(i) \times \frac{\sum_{j=1}^{i-1} \frac{ProductividadReal_Incr(j)}{EAF(j)}}{i-1} \quad (4.8)$$

Luego, se determinan los PFSA para cada incremento empleando los PFSA de cada uno de los casos de uso que fueron calculados en la actividad 3 y que van a ser implementados en ese incremento. La fórmula 4.9 representa este cálculo. En la fórmula, **PFSA_Incr(i)** corresponde a la cantidad de PFSA para el incremento “i”, **PFSA_CU(k)** son la cantidad de PFSA del caso de uso “k”, y “k” son los casos de uso que van a ser construidos en el incremento “i”.

$$PFSA_Incr(i) = \sum_{k=1}^n PFSA_CU(k) \quad (4.9)$$

Finalmente, se calcula el esfuerzo requerido para construir el siguiente incremento mediante la fórmula 4.10 que se muestra a continuación. En la fórmula **EsfuerzoEstimado_Incr(i)** es el esfuerzo estimado para el incremento “i”, **PFSA_Incr(i)** corresponde a la cantidad de PFSA para el incremento “i” y **ProductividadEstimada_Incr(i)** corresponde a la productividad estimada para el incremento “i”.

$$EsfuerzoEstimado_Incr(i) = PFSA_Incr(i) \times ProductividadEstimada_Incr(i) \quad (4.10)$$

Podría ocurrir que el esfuerzo estimado sea mayor o menor a la capacidad de construcción del equipo de desarrollo. Para estos casos, se tendría que modificar la cantidad de casos de uso que se pensaron construir para ese incremento. Cabe resaltar que la fórmula 4.10 es similar a la empleada por Hericko et. al [Hericko, 2008]. La diferencia radica en que se emplean puntos de función ajustados (PF) y no puntos de función sin ajustar.

4.2.3.3 Evaluar resultados de la construcción del incremento (actividad 6)

Luego de finalizar la construcción de un incremento se tienen que realizar evaluaciones y revisiones de lo ocurrido. En relación a la estimación y planificación de incrementos siguientes, se debe tener en cuenta que el contexto del proyecto puede cambiar (conocimiento de la plataforma de desarrollo, integración del equipo de desarrollo, etc), por lo que esto se tiene que considerar para estimar el esfuerzo de los incrementos que faltan desarrollar.

También, se debe determinar si es que se tienen que actualizar los ficheros (ILF/EIF) y transacciones (EI, EO y EQ) en caso de cambio de requisitos, y recalculando EM para calcular el nuevo EAF, propuesto por Cocomo II, en caso de cambios contextuales u organizacionales. En el caso que no se haya construido un caso de uso, este deberá ser reasignado a siguientes incrementos.

4.3 Ejemplo de la aplicación de Tupuy

El ejemplo consiste en un sistema de información para el área de ventas de una empresa. Este ejemplo es una simplificación de un caso real para que se pueda entender de manera más sencilla la aplicación de la técnica. Por ejemplo, no se ha incluido los casos de uso asociados a seguridad como “Login” y administración de usuarios. La Figura 4-30 muestra al lado izquierdo el diagrama de casos de uso del ejemplo y a la derecha, el diagrama de clases de análisis o modelo conceptual del sistema. La especificación de casos de uso se ha incluido en el Anexo B.

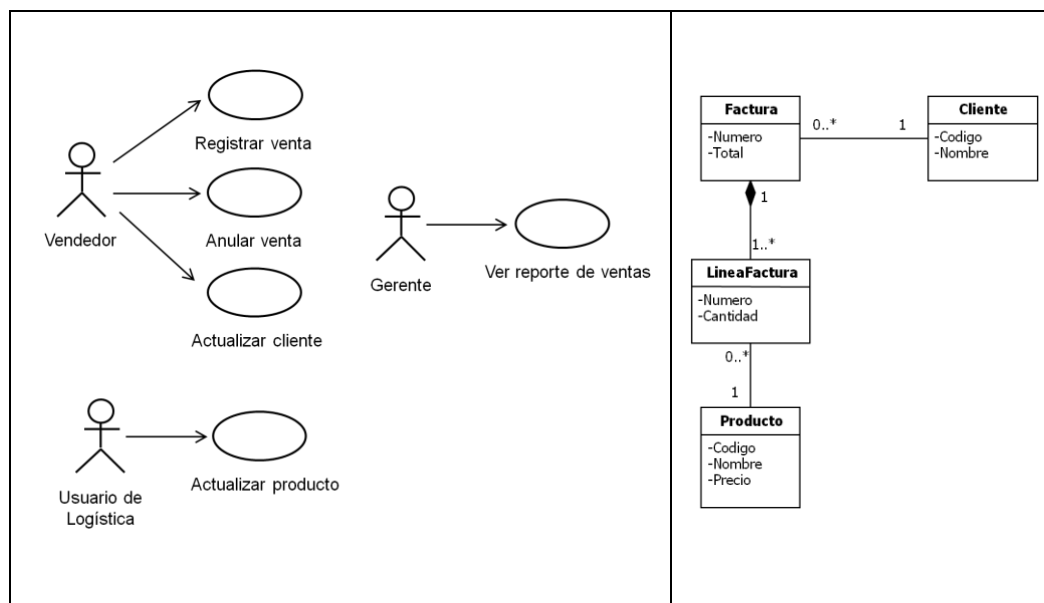


Figura 4-30. Diagramas de casos de uso y de clases para el ejemplo de aplicación de Tupuy

Como se puede observar en la Figura 4-30, el sistema de información consta de cinco casos de uso: Registrar venta, Anular venta, Actualizar cliente, Actualizar producto y Ver reporte de ventas. Tomando en cuenta la especificación de casos de uso, se realizó el diagrama de clases de análisis que se muestra a la derecha de la figura. Los patrones de Coad empleados y a que clases están relacionadas son:

- Patrón **Participante-Transacción**: Cliente-Factura
- Patron **Transacción-LíneaItemTransacción**: Factura-LineaFactura.
- Patrón **Item-LíneaItem**: Producto-LíneaItem

A continuación se muestran las actividades seguidas según Tupuy para poder definir los incrementos y estimar el esfuerzo del ejemplo mostrado

4.3.1 Aplicación de UML₂FP

La primera tarea que especifica UML₂FP consiste en determinar los ficheros con sus DET y RET. Para ello, se deberá emplear el diagrama de clases de análisis. La Figura 4-31 muestra el diagrama de clases de análisis del ejemplo y qué ficheros de PF con sus DET y RET están relacionados.

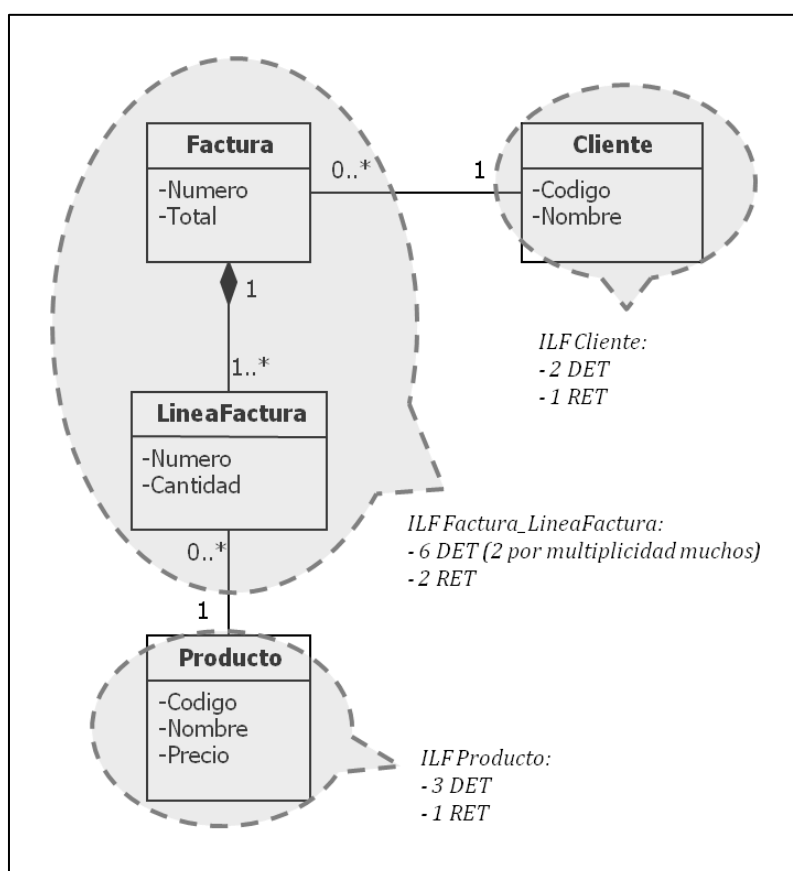


Figura 4-31. Clases y ficheros de Puntos de función relacionados del ejemplo

En la Figura 4-31, se puede observar que se han obtenido tres ficheros del diagrama de clases de análisis. La secuencia y las reglas seguidas (ver acápite 4.2.1.1) para determinarlos son las siguientes:

1. Para el conjunto de clases Factura-LineaFactura-Producto, se siguió la regla 1b. Con esta regla se determinó que hay dos ficheros: Producto

con 3 DET y 1RET, y Factura con 5 DET (un DET debido a la multiplicidad con Producto) y 2 RET.

2. Para la clase Cliente y el fichero Factura_LineaFactura, se siguió la regla 5. Luego se siguió la regla 2 para el conjunto de clases Cliente-Factura, como estas clases son independientes, Cliente se consideró como un fichero. Como existe una multiplicidad de muchos en la clase Factura, se le añadió un DET al fichero Factura_LineaFactura. Por lo tanto, el fichero Factura_LineaFactura pasa de tener 5 DET a 6 DET.
3. Para determinar si los ficheros son ILF o EIF, se siguió la regla 6. Según esta regla, los tres ficheros son ILF. De acuerdo a la especificación de casos de uso, los tres ficheros son mantenidos por el sistema que se está contando.
4. Finalmente, siguiendo la regla 6, se determinó la complejidad de cada fichero y su peso en PFSA. Para ello, se utilizó las tablas definidas por el Ifpug [Ifpug, 2004].

Luego de determinar la cantidad de PFSA de cada fichero, se tuvo que identificar las transacciones relacionadas a cada caso de uso. La Figura 4-32 muestra el caso de uso Actualizar producto y sus transacciones asociadas.

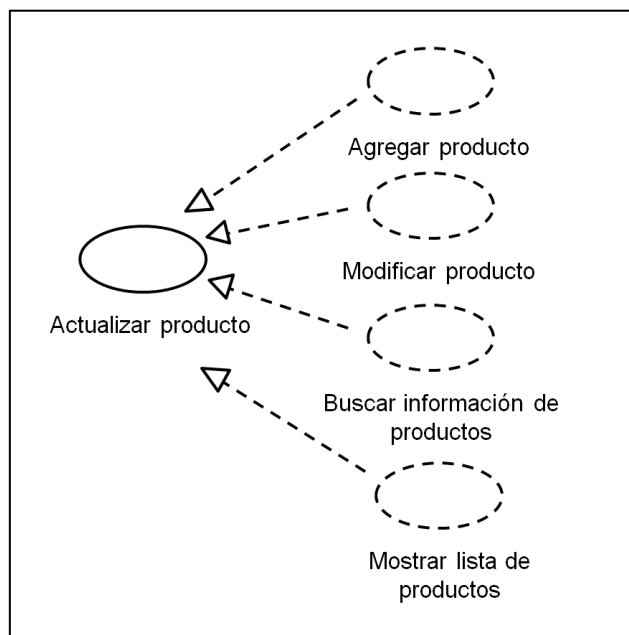


Figura 4-32. Casos de uso y transacciones de PF asociados del ejemplo

La Tabla 4-5 muestra las transacciones identificadas por cada caso de uso. En este caso, ninguna de las transacciones fue identificada más de una vez en todos los casos de uso. La primera columna muestra el nombre del caso de uso; la segunda columna, el nombre de la transacción; la tercera, el tipo de

transacción (EI, EO, EQ); y la última columna, la cantidad de PFSA de cada transacción.

Tabla 4-5. PFSA por transacción de cada caso de uso del ejemplo

Caso de Uso	Nombre Transacción	Tipo	PFSA
Actualizar producto	Agregar producto	EI	3
	Modificar producto	EI	3
	Mostrar información de producto	EQ	3
	Mostrar lista de productos	EQ	3
Actualizar cliente	Agregar cliente	EI	3
	Modificar cliente	EI	3
	Mostrar información de cliente	EQ	3
	Mostrar lista de clientes	EQ	3
Registrar venta	Calcular subtotal de línea	EO	4
	Registrar venta	EI	6
Anular venta	Buscar factura	EQ	4
	Anular venta	EI	3
Ver reporte de ventas	Ver reporte de ventas	EO	7

Para determinar el cálculo de PFSA se identificó el tipo de transacción, después su complejidad y finalmente su peso en PFSA. Todo esto se realizó siguiendo las reglas definidas por el Ifpug [Ifpug, 2009].

4.3.2 Aplicación de UCPD

La Figura 4-33 muestra el diagrama de precedencias o *Use Case Precedence Diagram (UCPD)* para el ejemplo. Para determinar el UCPD, no se empleó la regla 1 (ver subsección 4.2.2) porque ninguno de los casos de uso tenía precondiciones definidas.

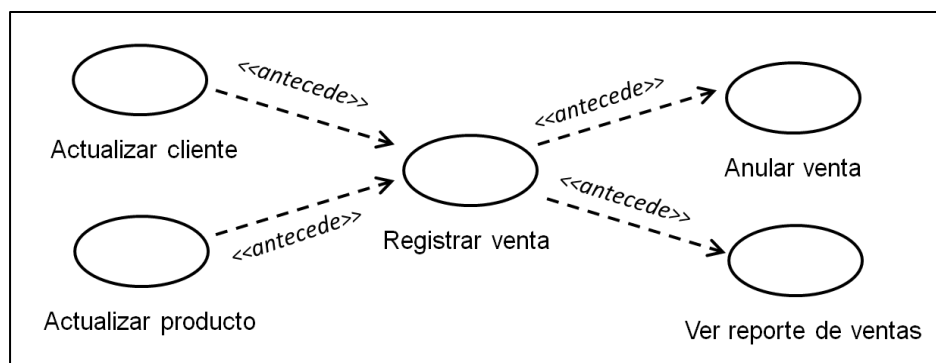


Figura 4-33. Diagrama UCPD del ejemplo

De acuerdo a UCPD, la secuencia de construcción debería considerar primero el desarrollo de los casos de uso que se encuentran a la izquierda y después los que se encuentran a la derecha. Para el ejemplo, como se puede apreciar en la Figura 4-33, primero se tendrían que construir los casos de uso Actualizar cliente y Actualizar producto. Estos dos tendrían que construirse antes que Registrar venta. Tras implementar Registrar venta se podrán construir Ver reporte de ventas y Anular venta.

4.3.3 Aplicación de Incremental-FP

Incremental-FP indica que se deben realizar tres actividades: estimar el tamaño de cada caso de uso (ver acápite 4.3.3.1), definir incrementos y esfuerzo requerido (ver acápite 4.3.3.2) y evaluar resultados de la construcción del incremento (ver acápite 4.3.3.3). A continuación se muestran esas actividades para el ejemplo que se ha propuesto.

4.3.3.1 Estimación del tamaño de cada caso de uso (actividad 3)

La Tabla 4-6 muestra los PFSA debido a los ficheros para el ejemplo. En la tabla, la primera columna muestra el nombre de los casos de uso; la segunda columna, el nombre de la transacción asociada a los casos de uso; desde la tercera hasta la quinta columna se muestra con una “x” que fichero es empleado por qué transacción; y la última columna presenta los PFSA de cada transacción debido a los ficheros.

Tabla 4-6. PFSA debido a ficheros por cada transacción del ejemplo

Caso de uso	Transacción	Fichero ILF			PFSA por fichero
		Producto	Cliente	Factura_LineaFactura	
Actualizar producto	Agregar producto	x			1.00
	Modificar producto	x			1.00
	Mostrar información de producto	x			1.00
	Mostrar lista de productos	x			1.00
Actualizar cliente	Agregar cliente		x		1.40
	Modificar cliente		x		1.40
	Mostrar información de cliente		x		1.40
	Mostrar lista de clientes		x		1.40
Registrar venta	Calcular subtotal de línea	x			1.00
	Registrar venta	x	x	x	4.15
Anular venta	Buscar factura			x	1.75

	Anular venta			x	1.75
Ver reporte de ventas	Ver reporte de ventas	x		x	2.75

Según Incremental-FP, la cantidad de PFSA de cada fichero se reparte equitativamente entre las transacciones que la emplean. Por ejemplo, el fichero Cliente es utilizado por 5 transacciones, como el peso en PFSA de ese fichero es 7, se reparte 1.40 PFSA a cada una de las transacciones que la emplean: Agregar cliente, Modificar cliente, Mostrar información de cliente, Mostrar lista de clientes y Registrar venta.

Luego de haber calculado los PFSA que tiene cada transacción debido a los ficheros y conociendo los PFSA propios de cada transacción, se pueden calcular los PFSA para cada caso de uso. La Tabla 4-7 muestra para el ejemplo; en la primera columna, los casos de uso; en la segunda columna, el nombre de las transacciones; en la tercera columna, el tipo de transacción; en la cuarta columna, los PFSA propios de cada transacción; en la quinta columna, los PFSA debido a los ficheros por cada transacción; y, finalmente, en la última columna, los PFSA de cada caso de uso. En esta última columna se han sumado, por cada caso de uso, los PFSA de sus transacciones asociadas.

Tabla 4-7. PFSA de cada caso de uso del ejemplo

Caso de uso	Transacción				PFSA por Caso de Uso
	Nombre	Tipo	PFSA por Transacción	PFSA por Ficheros	
Actualizar producto	Agregar producto	EI	3	1.00	16.00
	Modificar producto	EI	3	1.00	
	Mostrar información de producto	EQ	3	1.00	
	Mostrar lista de productos	EQ	3	1.00	
Actualizar cliente	Agregar cliente	EI	3	1.40	17.60
	Modificar cliente	EI	3	1.40	
	Mostrar información de cliente	EQ	3	1.40	
	Mostrar lista de clientes	EQ	3	1.40	
Registrar venta	Calcular subtotal de línea	EO	4	1.00	15.15
	Registrar venta	EI	6	4.15	
Anular venta	Buscar factura	EQ	4	1.75	10.50
	Anular venta	EI	3	1.75	
Ver reporte de ventas	Ver reporte de ventas	EO	7	2.75	9.75
Total					69.00

4.3.3.2 Definición de incrementos y esfuerzo requerido (actividad 4)

Tomando en cuenta el diagrama UCPD y considerando que se van a construir tres incrementos, se determinó qué caso de uso se iba a construir en qué incremento. La Tabla 4-8 muestra lo indicado.

Tabla 4-8. Casos de uso que se implementarán por cada incremento en el ejemplo

Caso de Uso	Incremento 1	Incremento 2	Incremento 3
Actualizar producto	x		
Actualizar cliente		x	
Registrar venta		x	
Anular venta			x
Ver reporte de ventas			x

Suponiendo que la productividad de proyectos previa, la cual podríamos llamarla productividad histórica, es de 3.50 horas-persona/PFSA para un EAF de 1.00; entonces podríamos estimar el esfuerzo que necesitaríamos para el primer incremento con la fórmula 4.6 (ver acápite 4.2.3.2). Para el ejemplo, y como suposición, consideramos que el equipo de desarrollo no tiene mucha experiencia en el lenguaje ni el desarrollo de aplicaciones; por lo tanto, el EAF sería de 1.46. Con estos datos, podríamos calcular la productividad estimada, la cual sería de 5.11 horas-persona/PFSA. Como se piensa implementar el caso de uso Actualizar producto y éste tiene 16 PFSA, entonces el esfuerzo estimado sería de 81.76 horas-persona (el esfuerzo es el resultado de la productividad por la cantidad de PFSA, según fórmula 4.10). Un resumen de lo explicado se puede ver en la Tabla 4-9.

Tabla 4-9. Estimación para el primer incremento del ejemplo

Incremento	PFSA	EAF	Productividad Estimada (h-p/PFSA)	Esfuerzo Estimado (h-p)	Productividad Real (h-p/PFSA)	Esfuerzo Real (h-p)
Histórico	-	1.00	-	-	3.50	-
1	16.00	1.46	5.11	81.76		

Suponiendo que el esfuerzo real empleado en el primer incremento fue de 85 horas-persona, se puede determinar que la productividad real fue de 5.31 horas-persona/PFSA para un EAF de 1.46. Como esta información es propia del mismo equipo, permitirá estimar con más precisión el esfuerzo necesario para el siguiente incremento. Suponiendo que en el primer incremento el equipo ganó experiencia en el desarrollo de aplicaciones y de la herramienta, entonces

el EAF cambiaría a 0.80. Con esta información se podrá determinar el esfuerzo requerido para el segundo incremento. La Tabla 4-10 incluye la estimación del esfuerzo para el segundo incremento, considerando que se va a construir Actualizar cliente y Registrar venta.

Tabla 4-10. Estimación para el segundo incremento del ejemplo

Incremento	PFSA	EAF	Productividad Estimada (h-p/PFSA)	Esfuerzo Estimado (h-p)	Productividad Real (h-p/PFSA)	Esfuerzo Real (h-p)
Histórico	-	1.00	-	-	3.50	-
1	16.00	1.46	5.11	81.76	5.31	85.00
2	32.75	0.80	2.91	95.33		

Suponiendo que al finalizar el primer incremento, se determina que el esfuerzo real empleado fue de 82 horas-persona, entonces la productividad real sería de 2.50 horas-persona/PFSA para un EAF de 0.80. Teniendo en cuenta que el contexto del proyecto no ha cambiado para el tercer incremento, es decir que el EAF sería de 0.80, entonces se puede estimar el esfuerzo estimado empleando la información de la productividad real del primer y del segundo incremento. Para ello, se utiliza la fórmula 4.9 (ver acápite 4.2.3.2). La Tabla 4-11 incluye la estimación del esfuerzo para el tercer incremento, considerando que se va a construir Anular venta y Ver reporte de ventas.

Tabla 4-11. Estimación para el tercer incremento del ejemplo

Incremento	PFSA	EAF	Productividad Estimada (h-p/PFSA)	Esfuerzo Estimado (h-p)	Productividad Real (h-p/PFSA)	Esfuerzo Real (h-p)
Histórico	-	1.00	-	-	3.50	-
1	16.00	1.46	5.11	81.76	5.31	85.00
2	32.75	0.80	2.91	95.33	2.50	82.00
3	20.25	0.80	2.71	54.82	-	-

4.3.3.3 Evaluación de resultados de la construcción del incremento (actividad 6)

En el acápite anterior (acápite 4.3.3.2), se ha incluido el caso en el que las estimaciones son cercanas al esfuerzo real, pero podría ocurrir que no se produzca de esa manera.

Por ejemplo, podría ocurrir que no se pudiera construir un caso de uso en uno de los incrementos. En ese caso se tendría que redefinir la secuencia de construcción de casos de uso que se indica en la Tabla 4-8 y asignar ese nuevo caso de uso en el siguiente incremento.

En el caso que la diferencia entre el esfuerzo estimado y real sea mayor a 20%, se debería evaluar el porqué de esta diferencia. Podría darse el caso que se haya producido un cambio en el contexto de la construcción del incremento, por ejemplo cambios de personal, que no fueron considerados al inicio del incremento. Para estos casos, se deberá tomar en cuenta el EAF actualizado que considere estos cambios para así poder estimar el esfuerzo requerido para construir el siguiente incremento.

Es muy probable que se puedan producir diferencias significativas entre el esfuerzo real y estimado del primer incremento, porque la información tomada podría corresponder a un proyecto en el que por ejemplo ha participado otro equipo de proyecto. No obstante, las estimaciones de los siguientes incrementos deberán ser muy cercanas a los esfuerzos reales, porque es información propia del mismo proyecto.

4.4 Evaluación de TUPUY

En los siguientes capítulos, se detalla la evaluación de las técnicas que forman parte de Tupuy. Las evaluaciones se han realizado de la siguiente manera:

1. Experimentos controlados en los que los participantes tuvieron que utilizar la técnica original de PF, la cual emplea diagramas E-R, y las reglas de UML₂FP, la cual emplea diagramas de clases, para poder determinar los ficheros y sus RET. El objetivo de estos experimentos fue determinar si UML₂FP permite determinar ficheros y sus RET con por lo menos la misma precisión que la técnica original de PF propuesta por el Ifpug. Esta evaluación se muestra en el capítulo 5.

Cabe resaltar que la parte más crítica en el cálculo de ficheros de PF es la identificación de ficheros y sus RET, ya que si esto no se hace apropiadamente, la variación en PFSA puede ser muy alta. Por ejemplo, si se identifica un fichero con 2 RET, generalmente esto correspondería a 7 PFSA; pero si en realidad esto correspondiera a 2 ficheros y cada uno con su RET, implicaría una cantidad de 14 PFSA, es decir el doble. Por ello, es que los experimentos controlados solo se centran en evaluar la identificación de ficheros y RET.

2. Experimentos controlados en los que los participantes tuvieron que emplear sus técnicas propias o ad hoc para determinar la secuencia de construcción de casos de uso y luego UCPD. Esta validación se muestra en el capítulo 6.

3. Estimación del esfuerzo de los incrementos en proyectos desarrollados por alumnos de pregrado. Esta validación tiene dos tipos: proyectos en los que los alumnos emplearon Incremental-FP para estimar el esfuerzo requerido para un siguiente incremento o proyectos en los que los alumnos registraron información para realizar la estimación del esfuerzo (casos de uso construido por incremento y esfuerzo real empleado) y con esa información se determinó al final del proyecto si la técnica hubiese dado buenos resultados si se hubiese empleado. Aunque esta evaluación principalmente está relacionada a Incremental-FP, cabe resaltar que los alumnos de pregrado también emplearon UCPD y UML₂FP para sus proyectos; por ende, esta evaluación correspondería también a todo Tupuy. Esta evaluación se muestra en el capítulo 7.

Evaluación de la Conversión a Ficheros de UML₂FP

Capítulo

5

Este capítulo tiene como finalidad evaluar la conversión a ficheros, a partir de un diagrama de clases de análisis, que se ha definido como parte de UML₂FP y que se detalla en el capítulo 4.

Para esta evaluación, se ejecutaron cuatro experimentos controlados: uno con profesionales, dos con alumnos de posgrado y uno con alumnos de pregrado. También, tras realizar y procesar los datos obtenidos en el primer experimento controlado, se pudo observar que uno de los casos no estaba adecuadamente descrito; por ello se realizó un estudio piloto con docentes y asistentes de docencia para corroborar si todos los casos o situaciones, en los que los participantes tenían que aplicar ambas técnicas, estaban correctamente descritas.

El capítulo se estructura como sigue: la Sección 5.1 presenta una visión general de la evaluación, la Sección 5.2 muestra los materiales empleados en los experimentos, la Sección 5.3 describe las tareas realizadas durante los experimentos, la Sección 5.4 detalla los resultados obtenidos en los cinco experimentos controlados y, finalmente, se presentan las conclusiones de esta evaluación.

5.1 Visión General de la Evaluación

En el año 2005, se desarrolló un conjunto de reglas para la conversión de ficheros en base a un diagrama de clases de análisis. Las reglas que se muestran en el capítulo 4 y que forma parte de UML₂FP, es una versión mejorada de las reglas definidas en el 2005 [Pow-Sang, 2007][Pow-Sang, 2008]. Aunque para esta técnica previa a UML₂FP también se consideraron los patrones de [Coad, 1997] para su definición, carecía de algunas definiciones adicionales que podrían permitir su aplicación a más casos que los representados por esos patrones. También, se realizaron experimentos controlados con profesionales y estudiantes de pregrado para validar las reglas que se proponían con buenos resultados.

Tomando en cuenta la experiencia obtenida en los experimentos de la técnica previa a UML₂FP, se realizó una familia de experimentos que consistió en cinco experimentos controlados: el primero de ellos se realizó con profesionales con experiencia en la industria, el segundo fue realizado con profesores y asistentes de docencia, el tercero con alumnos de pregrado y los dos últimos con estudiantes de posgrado. Aunque hubiese sido suficiente con realizar experimentos controlados con profesionales y estudiantes de posgrado,

se realizó un experimento controlado con estudiantes de pregrado con el fin de comparar resultados y así poder obtener mejores conclusiones sobre la técnica y sobre los experimentos entre personas con diferentes niveles de experiencia en desarrollo de software.

La Tabla 5-1 muestra lo siguiente: la secuencia de ejecución de los experimentos, a quienes se aplicó, el año de realización y el acrónimo que se empleará en el resto del capítulo para identificarlos.

Tabla 5-1. Relación de experimentos controlados empleados para evaluar la conversión a ficheros de UML₂FP

Secuencia	Aplicado a	Año de realización	Acrónimo
1	Profesionales	2008	EC1-PROF
2	Docentes y asistentes de docencia (estudio piloto para determinar si la descripción de situaciones era correcta)	2009	EP2-DOC
3	Estudiantes de pregrado	2009	EC3-PREG
4	Estudiantes de posgrado del Curso de Especialización en Ing. de Software	2009	EC4-POST
5	Estudiantes de posgrado de la Maestría en Informática	2010	EC5-POST

Utilizando el criterio "PICOC" propuesto por Petticrew y Roberts [Petticrew, 2006], los experimentos controlados se definieron de la siguiente manera:

Población (*population*): estimación de proyectos software

Intervención (*intervention*): técnica UML₂FP

Comparación (*comparison*): técnica propuesta por Ifpug

Salidas (*outcomes*): por lo menos igual precisión para determinar los ficheros con sus RET

Contexto (*context*): profesionales y estudiantes de pregrado y posgrado de la Pontificia Universidad Católica del Perú.

La pregunta de investigación es ¿la precisión que se obtiene al aplicar UML₂FP para identificar ficheros es por lo menos similar que al aplicar la técnica definida por Ifpug? La variable independiente es la técnica empleada por los sujetos de experimentación para identificar los ficheros lógicos. La variable dependiente es la precisión; es decir, el grado de acercamiento entre los resultados medidos y el valor verdadero.

Se consideró como “valor verdadero”, los resultados que se deberían obtener al aplicar la técnica propuesta por Ifpug. Por ello, se utilizaron casos de estudio similares a los incluidos en la versión 4.2.1 y 4.2.3 del manual de PF [Ifpug, 2004] [Ifpug, 2009].

5.2 Materiales Empleados en los Experimentos

Los materiales empleados en los experimentos controlados fueron la descripción de los casos con sus diagramas E-R, la descripción de los casos con sus diagramas de clases de análisis, formularios para registrar el número de ficheros lógicos con sus RET por cada caso y un cuestionario para conocer la opinión de los estudiantes sobre la claridad de cada técnica. Los participantes tuvieron que identificar los ficheros y sus RET a siete casos. La Figura 5-1 muestra como ejemplo una parte del documento que contenía los casos o situaciones para la técnica original en el que incluían los diagramas E-R.

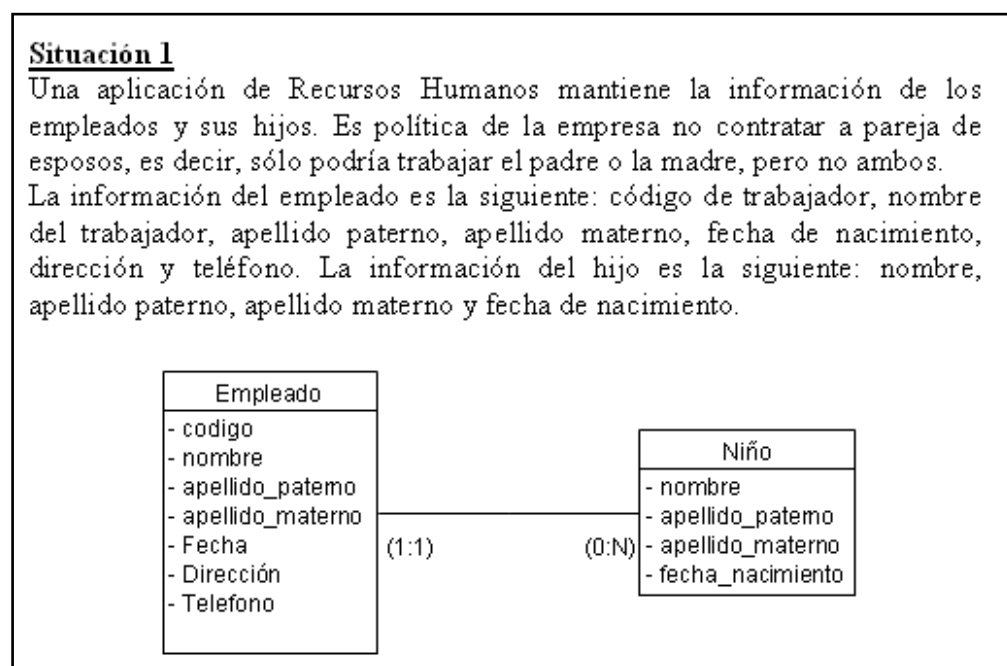


Figura 5-1. Vista parcial del documento que contenía los casos con los diagramas E-R

Cabe resaltar que de los siete casos empleados para que los alumnos apliquen la técnica del Ifpug (con diagramas E-R), seis fueron diferentes a los siete que tuvieron que utilizar con UML₂FP (con diagramas de clases). Adicionalmente, uno de los casos utilizados fue el mismo para ambas técnicas, el cual no se consideró en las pruebas estadísticas porque no se puede garantizar

que los resultados obtenidos en este caso hayan sido influenciados por el efecto de aprendizaje. En cuanto al cuestionario, su objetivo fue conocer la percepción de los participantes sobre la técnica propuesta. En el anexo C se encuentra el material que se utilizó en los experimentos.

5.3 Tareas Realizadas Durante los Experimentos

Para los experimentos controlados, se seleccionó el diseño de experimento intrasujeto, es decir que los estudiantes tuvieron que determinar los ficheros y sus RET para los casos y situaciones definidas con la técnica definida por el Ifpug y con UML₂FP y la del Ifpug. La Figura 5-2 muestra el diseño del experimento.

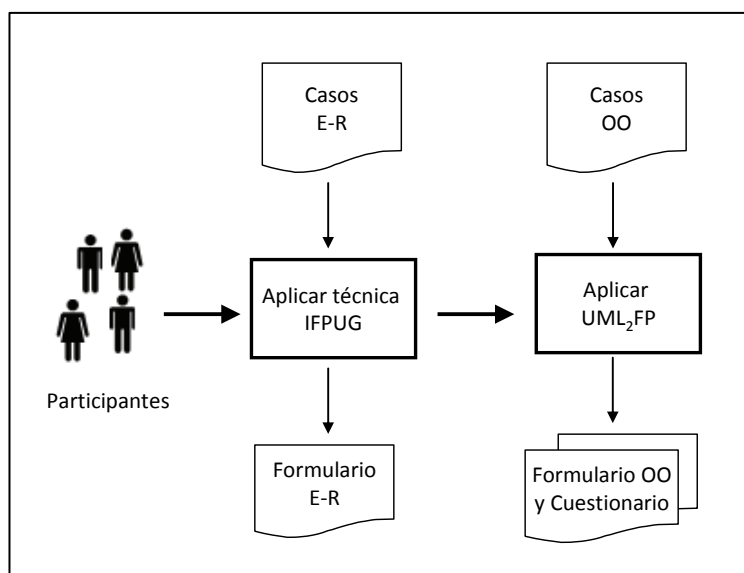


Figura 5-2. Diseño de los experimentos controlados para la conversión a ficheros de UML₂FP

Como se puede observar en la Figura 5-2, los alumnos tuvieron que aplicar la técnica de PF del Ifpug y luego la UML₂FP. La Tabla 5-2 muestra las tareas que fueron realizadas por los participantes en el experimento. La primera columna presenta la secuencia de ejecución de la tarea y la segunda columna, la descripción de la tarea.

Tabla 5-2. Tareas realizadas por los participantes de los experimentos controlados para la conversión a ficheros UML₂FP

Nº	Descripción de la Tarea
1	Recibir situaciones E-R, explicación de la técnica del Ifpug y formulario para registrar los ficheros lógicos identificados con sus RET
2	Identificar ficheros lógicos según las situaciones y sus diagramas E-R
3	Entregar formulario completado
4	Recibir situaciones OO, explicación de la técnica UML ₂ FP y formulario para registrar los ficheros lógicos identificados con sus RET
5	Identificar ficheros lógicos
6	Entregar formulario completado y recibir cuestionario
7	Entregar cuestionario completado

La sesiones duraron aproximadamente 90 minutos, incluyendo la revisión de la técnica de PF si es que era necesario, y los participantes emplearon 60 minutos en promedio para completar todas las tareas.

En los experimentos controlados que se emplearon para evaluar la técnica previa a UML₂FP, se incluyeron como tareas el realizar los diagramas E-R y los diagramas de clases de análisis de los casos antes de aplicar ambas técnicas. Durante estas tareas, se tuvo que asesorar a los participantes para evitar que hicieran mal los diagramas, a pesar que todos tenían conocimiento de cómo realizarlos. Es por ello, que en los experimentos controlados se les entregaron todos los casos con sus diagramas respectivos (E-R y clases). Es más, el objetivo del experimento es determinar que tan precisas son las técnicas para identificar ficheros y el eliminar esas tareas no afectaría el objetivo de investigación propuesto.

5.4 Resultados

Dado que los casos empleados para cada técnica no fueron los mismos, éstos se clasificaron según el tipo de relación que debería emplearse en un diagrama de clases: asociación, agregación, composición, generalización o clase asociación; con el fin de comparar los resultados obtenidos. Como en los diagramas E-R no se emplean relaciones de composición ni de agregación, se clasificaron en una categoría denominada **composición/agregación** a aquellos casos en los que se emplearía ese tipo de relaciones si se utilizara un diagrama de clases de análisis en vez de un diagrama E-R.

Para ambas técnicas, se emplearon dos casos para la categoría de asociación, un caso para clase asociación, un caso para generalización y tres casos para composición/agregación. En esta última categoría, uno de los tres casos utilizados fue el mismo para ambas técnicas; por ello, no se consideraron los resultados de ese caso, para eliminar la amenaza a la validez interna relacionada a efectos de aprendizaje.

Por cada caso, se calificó con un “1” (uno) si el estudiante identificó correctamente el número de ficheros lógicos y RET, y “0” (cero) si el participante no lo identificó correctamente. Esto quiere decir que para la categoría de asociación, el alumno podría obtener un máximo de “2”, ya que son dos casos para este tipo de relación; un máximo de “2” en la categoría de composición/agregación, ya que son dos casos para este tipo de relación (no se consideró el caso que fue el mismo para las dos técnicas, porque sino tendría que ser un máximo de “3”); un máximo de “1” para la categoría de clase asociación, ya que solo fue un caso; y un máximo de “1” para la categoría de generalización, ya que fue un solo caso.

La Tabla 5-3 muestra un ejemplo de cómo se mostrarán los resultados que se obtuvieron en cada experimento controlado. Por categoría, la primera columna muestra los resultados obtenidos con la técnica del Ifpug, que emplea diagramas E-R, y la segunda columna, con la UML₂FP, que emplea diagramas de clases.

Tabla 5-3. Ejemplo de resultados obtenidos por categoría en experimentos controlados para evaluar UML₂FP

Participante	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
1	1	1	1	1	1	1	1	1
2	1	0	1	1	1	0	1	0
3	2	1	2	2	0	1	1	0
4	2	1	2	2	0	0	0	1
5	1	1	2	2	0	1	0	1

Debido a que las reglas para las categorías de asociación, clase asociación y generalización son similares entre la técnica del Ifpug y UML₂FP, las hipótesis estadísticas formuladas para probar ambas técnicas fueron:

$$H_0: \mu_1 = \mu_2, \quad \alpha = 0.05$$

$$H_a: \mu_1 \neq \mu_2$$

Donde μ_1 es la media del cantidad de aciertos obtenido con la técnica del Ifpug y μ_2 , con UML₂FP, ambas para la misma categoría. α representa el 5% de significación, esto quiere decir que existe un 0.05 de probabilidad de aceptar la hipótesis alternativa cuando la hipótesis nula es verdadera (error de tipo I). La herramienta que se empleó para el análisis estadístico de los experimentos controlados incluidos en esta sección fue XLStat 2009.

5.4.1 EC1-PROF: Primer experimento controlado con profesionales

En este experimento participaron dieciséis profesionales con al menos dos años de experiencia en proyectos de desarrollo de software. Muchos de ellos trabajaban en compañías que están mejorando sus procesos de desarrollo de software. La Tabla 5-4 presenta el detalle de los resultados obtenidos por cada participante.

Tabla 5-4. Resultados obtenidos por cada participante en EC1-PROF

Participante	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
1	1	1	1	1	1	1	1	1
2	1	0	1	1	1	0	1	0
3	2	1	2	2	0	1	1	0
4	2	1	2	2	0	0	0	1
5	1	1	2	2	0	1	0	1
6	2	1	2	2	1	1	1	1
7	2	2	2	2	1	1	1	1
8	2	1	2	2	0	0	1	1
9	2	1	2	2	1	1	1	1
10	2	1	1	2	0	1	1	1
11	1	0	2	2	1	1	1	1
12	1	2	2	2	1	1	1	1
13	1	1	2	2	0	1	1	1
14	1	2	2	2	1	1	0	0
15	2	1	2	2	1	1	1	0
16	2	1	2	2	1	1	1	1

La Tabla 5-5 presenta la información estadística obtenida luego de procesar los datos contenidos en la Tabla 5-4. Por categoría, la primera columna muestra los resultados obtenidos con la técnica del Ifpug y la segunda columna, con UML₂FP.

Tabla 5-5. Valores estadísticos de los resultados obtenidos en EC1-PROF

Variable	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
Observaciones	16	16	16	16	16	16	16	16
Mínimo	1	0	1	1	0	0	0	0
Máximo	2	2	2	2	1	1	1	1
Media	1.563	1.063	1.813	1.875	0.625	0.813	0.813	0.750
Desv. Estándar	0.512	0.574	0.403	0.342	0.500	0.403	0.403	0.447

Como se puede observar en la tabla anterior, las medias que se obtuvieron con las dos técnicas son similares. Sin embargo, en la categoría de asociación, se puede notar que hay una diferencia mayor; obteniéndose mejores resultados con la técnica del Ifpug, técnica que emplea diagramas E-R, que con UML₂FP, técnica que emplea diagramas de clases de análisis. Para obtener resultados más concluyentes, se tuvieron que aplicar más pruebas estadísticas, las cuales se detallan a continuación.

Antes de poder utilizar pruebas estadísticas para comparar las medias obtenidas para las categorías de asociación y de composición/agregación, se tiene que establecer si estas muestras siguen una distribución normal. Para ello, se aplicó la prueba de Shapiro-Wilk. La Tabla 5-6 muestra los resultados obtenidos con este tipo de prueba. Para las categorías restantes, no se aplicó este tipo prueba, porque los valores que podrían tomar es cero o uno; por lo tanto no siguen una distribución normal.

Tabla 5-6. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC1-PROF

Variable	Asociación		Composición/ Agregación	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
W	0.638	0.748	0.484	0.398
p-valor	<0.0001	0.001	<0.0001	<0.0001

Dado que el p-valor computado fue menor que el nivel de significación establecido ($\alpha = 0.05$), se rechaza la hipótesis de distribución normal para todas las técnicas. Debido a resultados obtenidos, no se pudieron utilizar técnicas paramétricas para probar esos resultados. Por ello, se utilizó la prueba no paramétrica de los rangos señalados de Wilcoxon, pero como esta prueba emplea la mediana y no la media, se considerarán las hipótesis estadísticas definidas al inicio de esta sección cambiando la media (μ) por mediana (ver inicio de la Sección 5.4)

Tabla 5-7. Resultados de la prueba de los rangos señalados de Wilcoxon en EC1-PROF por categoría

Variable	Asociación	Clase asociación	Generalización	Composición /agregación
V	105.000	14.000	42.000	0.000
Valor esperado	63.000	35.000	35.000	8.0000
Varianza (V)	330.750	245.000	245.000	64.000
p-valor (dos colas)	0.011	0.190	0.678	0.174

Como se puede observar en la Tabla 5-7, el p-valor calculado para las categorías de clase asociación, generalización y composición/agregación fue mayor al nivel de significación $\alpha = 0.05$. Esto quiere decir que se puede afirmar, con un riesgo de equivocarse igual a 5%, que no hay diferencias significativas entre los resultados obtenidos con la técnica propuesta por Ifpug y con UML₂PF

Sin embargo, para la categoría de asociación, el p-valor calculado fue menor al nivel de significación $\alpha = 0.05$. Esto quiere decir, con un riesgo de equivocarse igual a 5%, que sí hay diferencias, cuando se pensaba que no debería haber; porque ambas técnicas proponen reglas similares. Por ello, se cambiaron las hipótesis estadísticas para determinar cuál de las dos técnicas ofrecía mejores resultados. Las hipótesis estadísticas definidas fueron:

$$H_0: Med_1 \leq Med_2, \quad \alpha = 0.05$$

$$H_a: Med_1 > Med_2$$

Donde Med_1 es la mediana de cantidad de aciertos obtenido con la técnica del Ifpug y Med_2 , con la técnica propuesta, ambas para la misma categoría. Para estas hipótesis, el p-valor obtenido fue 0.0055; por ello, se puede afirmar que para esta categoría, se obtuvieron mejores resultados con la técnica del Ifpug. Debido a que las técnicas son similares para esta categoría, se esperaba que no hubiera diferencias significativas en los resultados obtenidos por ambas.

En conclusión, para esta primera ejecución, se pudo comprobar estadísticamente que ambas técnicas producen los mismos resultados para los casos de las categorías composición/agregación, clase asociación y generalización. Sin embargo, para la categoría asociación, se obtienen mejores resultados con Ifpug; cuando se esperaba que se obtuvieran resultados parecidos, ya que ambas técnicas son similares.

5.4.2 EP2-DOC: Experimento piloto con docentes y asistentes de docencia

Debido a los resultados obtenidos en el primer experimento EC1-PROF, se realizó un estudio piloto con profesores y jefes de práctica para determinar si había algún problema con la descripción de los casos, en especial con los empleados para la categoría de asociación. La forma de ejecución fue similar a un experimento controlado. En este estudio piloto, se intercambiaron las situaciones utilizadas en EC1-PROF (experimento controlado con profesionales), de tal manera que los casos empleados en E-R se adaptaron para OO y los de OO se adaptaron para E-R. La Tabla 5-8 presenta el detalle de los resultados obtenidos por cada participante.

Tabla 5-8. Resultados obtenidos por cada participante en EC2-DOC

Participante	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
1	1	2	2	2	1	1	1	1
2	1	2	1	2	1	1	1	1
3	2	2	1	2	1	1	1	1
4	1	2	1	2	1	1	1	1

La Tabla 5-9 muestra la información estadística obtenida luego de procesar los datos contenidos en la Tabla 5-8. Como se puede observar en la tabla, las medias que se obtuvieron con las dos técnicas son diferentes para las categorías de asociación y composición/agregación. Los resultados en la categoría asociación son contradictorios a los obtenidos en la primera ejecución (ver sección 5.4.1). Por ello, se revisaron las situaciones de esta categoría y se conversó con los participantes del estudio. Se pudo concluir que al caso 1 le faltaba más información.

Tabla 5-9. Valores estadísticos de los resultados obtenidos en EC2-DOC

Variable	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
Observaciones	4	4	4	4	4	4	4	4
Mínimo	1	2	1	2	1	1	1	1
Máximo	2	2	2	2	1	1	1	1
Media	1.25	2	1.25	2	1	1	1	1
Desv. Estándar	0.5	0	0.5	0	0	0	0	0

La Tabla 5-10 muestra el caso 1 antes y después de la modificación. La columna de la izquierda presenta el caso 1 que se empleó en el experimento con profesionales EC1-PROF y la columna de la derecha muestra la misma situación a la que se le ha añadido un texto aclaratorio que se encuentra resaltado en gris y subrayado. En cuanto al diagrama empleado, la columna de la izquierda presenta un diagrama de clases y el de la derecha, un diagrama E-R.

Tabla 5-10. Situación 1 que se empleó originalmente en EC1-PROF y su actualización.

Situación Original Empleada en EC1-PROF (Diag. Clases)	Situación Modificada para Emplear en Sigüientes Experimentos (Diag. E-R)
<p>Una compañía decide realizar un programa de “adopción” de niños, de manera que un trabajador pueda “adoptar” la cantidad de niños que quiera para darles juguetes en Navidad y en su cumpleaños. Un niño no puede ser adoptado por más de un trabajador y todos los niños siempre deberán tener un padre o madre adoptivo.</p> <p>La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha de nacimiento, dirección y teléfono. La información del niño adoptado es la siguiente: nombre, apellido paterno, apellido materno, dirección y fecha de nacimiento.</p> <pre> classDiagram class Empleado { -codigo -nombre -apellido_paterno -apellido_materno -fecha -direccion -telefono } class NiñoAdoptado { -nombre -apellido_paterno -apellido_materno -direccion -fecha_nacimiento } Empleado "1" -- "0..*" NiñoAdoptado </pre>	<p>Una compañía decide realizar un programa de “adopción” de niños, de manera que un trabajador pueda “adoptar” la cantidad de niños que quiera para darles juguetes en Navidad y en su cumpleaños. Un niño no puede ser adoptado por más de un trabajador y todos los niños siempre deberán tener un padre o madre adoptivo. <u>Si a un empleado se le retira de la empresa, el o los niños que fueron adoptados por ese empleado son asignados a otra persona.</u></p> <p>La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha de nacimiento, dirección y teléfono. La información del niño adoptado es la siguiente: nombre, apellido paterno, apellido materno, dirección y fecha de nacimiento.</p> <pre> classDiagram class Empleado { -codigo -nombre -apellido_paterno -apellido_materno -fecha -direccion -telefono } class NiñoAdoptado { -nombre -apellido_paterno -apellido_materno -direccion -fecha_nacimiento } Empleado "1:1" -- "(0:N)" NiñoAdoptado </pre>

5.4.3 EC3-PREG: Tercer experimento controlado con estudiantes de pregrado

Debido a la experiencia obtenida en el segundo experimento EC2-DOC y luego de haber actualizado las situaciones, se realizó una tercera ejecución con estudiantes de pregrado. Cabe resaltar que se intercambiaron los casos que originalmente se utilizaron en el primer experimento realizado con profesionales EC1-PROF. Esto quiere decir que los casos que se utilizaron para aplicar la técnica del Ifpug en ese experimento controlado, se adaptaron para orientación a objetos, de tal manera que puedan ser empleadas con UML₂FP y viceversa.

Los estudiantes de pregrado que participaron en el experimento pertenecían al cuarto año de la carrera de Ingeniería Informática de la Pontificia Universidad Católica del Perú (PUCP) del semestre 2009-2, quienes estaban matriculados en la asignatura Ingeniería de Software. En esta asignatura, se les enseña a los alumnos técnicas de estimación y planificación de proyectos de software. La Tabla 5-11 muestra los conocimientos y experiencias de estos estudiantes.

Tabla 5-11. Conocimientos y experiencia de los estudiantes de pregrado que participaron en EC3-PREG

Características	Conocimiento y/o experiencia
Lenguajes de programación	Java, C#, Pascal, C and Prolog.
Técnicas de modelado de base de datos	Diagramas Entidad-Relación, , IDEF 1X
Técnicas de análisis y diseño	Estructurado y orientado a objetos
Gestión de Proyectos	Experiencia en gestión de pequeños proyectos de desarrollo que incluían proyectos de programación (equipos de 3 ó 4 estudiantes). No tenían experiencia previa en estimación.

La Tabla 5-12 presenta el detalle de los resultados obtenidos por cada participante. Por categoría, la primera columna muestra los resultados obtenidos con la técnica del Ifpug, que emplea diagramas E-R, y la segunda columna, con la UML₂PF, que emplea diagramas de clases.

Tabla 5-12. Resultados obtenidos por cada participante en EC3-PREG

Participante	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
1	2	1	1	2	0	0	0	0
2	2	2	1	2	1	1	1	1
3	1	1	1	1	0	0	1	1
4	2	1	1	2	0	0	1	1
5	1	2	1	1	0	1	0	0
6	2	1	1	2	0	0	1	1
7	1	2	0	1	0	0	0	0
8	1	1	2	2	1	1	1	1
9	2	1	0	1	0	0	0	0
10	2	1	1	2	1	1	0	0
11	2	1	1	1	1	1	0	0
12	1	2	2	2	1	1	1	1
13	2	1	1	1	1	1	0	0

Participante	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
14	2	2	1	1	1	1	1	1
15	1	2	1	2	0	0	0	1
16	2	1	1	2	0	1	1	1
17	1	2	1	2	1	1	1	1
18	2	1	1	2	1	1	1	1
19	2	2	2	2	1	1	1	1
20	1	2	1	2	1	1	1	1
21	2	2	1	2	1	1	1	1
22	2	2	1	2	1	1	1	1

En la Tabla 5-13 se muestra la información estadística obtenida luego de procesar los datos contenidos en la Tabla 5-12.

Tabla 5-13. Valores estadísticos de los resultados obtenidos en EC3-PREG

Variable	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
Observaciones	22	22	22	22	22	22	22	22
Mínimo	1	1	0	1	0	0	0	0
Máximo	2	2	2	2	1	1	1	1
Media	1.636	1.5	1.045	1.682	0.591	0.682	0.636	0.682
Desv. Estándar	0.492	0.592	0.486	0.477	0.503	0.477	0.492	0.477

Según los resultados obtenidos y que se muestran en la Tabla 5-13, las medias que se obtuvieron con las dos técnicas son similares. Sin embargo, en la categoría de composición/agregación, se puede observar que existe diferencias mayores que con las otras categorías y la media obtenida con UML₂FP es mejor que con la técnica propuesta por el Ifpug. Para obtener resultados más concluyentes, se tuvieron que aplicar más pruebas estadísticas las cuales se muestran a continuación.

Se utilizó la prueba estadística de Shapiro-Wilk para determinar si las muestras obtenidas para las categorías de asociación, composición/agregación, siguen una distribución normal. En el caso de las categorías restantes, no es necesario aplicar esta prueba porque como los valores podrían tomar valores de 0 ó 1, no siguen una distribución normal. La Tabla 5-14 muestra los resultados obtenidos con este tipo de prueba.

Tabla 5-14. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC3-PREG

Variable	Asociación		Composición/ Agregación	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
W	0.613	0.640	0.661	0.590
p-valor	<0.0001	<0.0001	<0.0001	<0.0001

Según los resultados obtenidos que se muestran en la tabla anterior, como el p-valor obtenido en todas las muestras es menor que el nivel de significación establecido ($\alpha = 0.05$), se puede afirmar que las muestras no siguen una distribución normal. Por ello, se tuvo que aplicar la prueba no paramétrica de los rangos señalados de Wilcoxon.

Tabla 5-15. Resultados de la prueba de los rangos señalados de Wilcoxon en EC3-PREG por categoría excepto composición/agregación

Variable	Asociación	Composición / agregación	Generalización	Clase asociación
V	135	0	0	0
Valor esperado	112.5	108.500	21.5	11
Varianza (V)	843.75	840.875	231.125	121
p-valor (dos colas)	0.449	< 0.0001	0.678	0.340

Como se puede observar en la Tabla 5-15, el p-valor calculado para las categorías de asociación, de clase asociación y de generalización fue mayor al nivel de significación $\alpha = 0.05$. Esto quiere decir que se puede afirmar, con un riesgo de equivocarse igual a 5%, que no hay diferencias significativas entre los resultados obtenidos con Ifpug y UML₂FP. Los resultados obtenidos son comprensibles, ya que para estas categorías, ambas técnicas son similares. Según los resultados presentados en la Tabla 5-15, el p-valor calculado la categoría de composición/agregación es menor que el nivel de significación de 0.05. Por lo tanto, podemos rechazar la hipótesis nula en este caso. Esto quiere decir que hay diferencias significativas en los resultados obtenidos en ambas técnicas. Por ello, se cambiaron las hipótesis estadísticas para determinar cuál de las dos técnicas ofrecía mejores resultados. Las hipótesis estadísticas definidas fueron:

$$H_0: \text{Med}_1 \geq \text{Med}_2, \quad \alpha = 0.05$$

$$H_a: \text{Med}_1 < \text{Med}_2$$

Donde Med_1 es la mediana de cantidad de aciertos obtenido con la técnica del Ifpug y Med_2 , con la técnica propuesta, ambas para la misma categoría. Para

estas hipótesis, el p-valor obtenido fue menor a 0.00001; por ello, se puede afirmar, con un riesgo de equivocarse igual a 5%, que UML₂FP produjo mejores resultados que la técnica del Ifpug para los casos de composición/agregación en el experimento controlado con estudiantes de pregrado EC3-PREG.

5.4.4 EC4-POST: Cuarto experimento controlado con estudiantes de posgrado

El experimento realizado a estudiantes de pregrado EC3-PREG fue replicado con profesionales quienes eran alumnos del Curso de Especialización de Ing. de Software de la PUCP del 2009. Aunque estos estudiantes de posgrado tenían al menos dos años de experiencia en proyectos de software y aunque utilizan herramientas de desarrollo OO en el trabajo, la mayoría de ellos no aplicaban técnicas de análisis y diseño OO o tenían poca experiencia en este tipo de técnicas. Sin embargo, una de las asignaturas dentro del curso de especialización consistía en una introducción sobre análisis y diseño OO, el cual tomaron previamente a la ejecución de este experimento. La Tabla 5-16 presenta el detalle de los resultados obtenidos por cada participante.

Tabla 5-16. Resultados obtenidos por cada participante en EC4-POST

Participante	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
1	2	1	1	2	1	1	0	0
2	2	1	1	2	0	1	1	1
3	2	2	1	1	1	0	0	1
4	2	1	1	1	1	1	0	0
5	2	2	1	2	0	1	1	1
6	1	2	2	2	1	1	0	0
7	1	2	1	2	0	0	0	0
8	2	1	2	2	1	1	0	0
9	1	2	1	2	0	1	0	1
10	0	2	0	2	0	1	1	1
11	0	1	0	2	0	1	1	1
12	2	0	1	1	0	1	0	1
13	0	1	0	0	0	0	1	0
14	1	1	1	1	1	1	0	0

Luego de procesar los datos de la Tabla 5-16, se obtuvo la información estadística que se presenta en la Tabla 5-17.

Tabla 5-17. Valores estadísticos de los resultados obtenidos en EC4-POST

Variable	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
Observaciones	14	14	14	14	14	14	14	14
Mínimo	0	0	0	0	0	0	0	0
Máximo	2	2	2	2	1	1	1	1
Media	1.285	1.357	0.929	1.571	0.429	0.786	0.357	0.500
Desv. Estándar	0.825	0.633	0.616	0.646	0.514	0.426	0.497	0.519

De manera similar a los resultados obtenidos con los alumnos de pregrado, como se puede observar en la tabla anterior (Tabla 5-17), las medias de las dos técnicas son muy parecidas en casi todas las categorías. Sin embargo, en la categoría de composición/agregación, se puede notar que hay diferencias y los resultados son mejores con UML₂FP que con la técnica del Ifpug. Para obtener resultados más concluyentes, se tuvieron que aplicar más pruebas estadísticas, las cuales se detallan a continuación.

Inicialmente, se aplicó la prueba de Shapiro-Wilk para determinar si las muestras de las categorías asociación y de composición/agregación siguen una distribución normal. La Tabla 5-18 muestra los resultados obtenidos con este tipo de prueba. Para las categorías restantes no se aplicó esta prueba, porque los valores que podrían tomar es cero o uno, por lo tanto no siguen una distribución normal.

Tabla 5-18. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC4-POST

Variable	Asociación		Composición/ Agregación	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
W	0.767	0.771	0.779	0.688
p-valor	0.002	0.002	0.003	0.000

Dado que el p-valor calculado fue menor que el nivel de significación establecido ($\alpha = 0.05$) se rechaza la hipótesis de distribución normal para todas las técnicas. Debido a resultados obtenidos, no se puede utilizar técnicas paramétricas para probar esos resultados. Es por ello, que se utilizó la prueba no paramétrica de los rangos señalados de Wilcoxon.

Tabla 5-19. Resultados de la prueba de los rangos señalados de Wilcoxon en EC4-POST por categoría excepto composición/agregación

Variable	Asociación	Composición /agregación	Clase asociación	Generalización
V	45.500	0	11.000	12.500
Valor esperado	49.500	38.500	38.500	25.000
Varianza (V)	235.125	216.125	211.750	156.250
Valor-P (dos colas)	0.819	0.01	0.064	0.337

Como se puede observar en la Tabla 5-19, el p-valor calculado para las categorías de asociación, de clase asociación y de generalización fue mayor al nivel de significación $\alpha = 0.05$. Esto quiere decir que se puede afirmar, con un riesgo de equivocarse igual a 5%, que no hay diferencias significativas entre los resultados obtenidos con Ifpug y la técnica propuesta. Los resultados obtenidos son comprensibles, ya que para estas categorías, ambas técnicas son similares. Sin embargo, el p-valor calculado la categoría de composición/agregación es menor que el nivel de significación de 0.05. Por lo tanto, podemos rechazar la hipótesis nula en este caso. Esto quiere decir que hay diferencias significativas en los resultados obtenidos en ambas técnicas. Por ello, se cambiaron las hipótesis estadísticas para determinar cuál de las dos técnicas ofrecía mejores resultados. Las hipótesis estadísticas definidas fueron:

$$H_0: Med_1 \geq Med_2, \quad \alpha = 0.05$$

$$H_a: Med_1 < Med_2$$

Donde Med_1 es la mediana de cantidad de aciertos obtenido con la técnica del Ifpug y Med_2 , con la técnica propuesta, ambas para la categoría de composición/agregación.

Tabla 5-20. Resultados de la prueba de los rangos señalados de Wilcoxon en composición/agregación en EC4-POST

Variable	Composición/agregación
V	0
Valor esperado	38.500
Varianza (V)	216.125
p-valor (una cola)	0.005

Según los resultados presentados en la tabla anterior, Tabla 5-20, el p-valor calculado es menor que el nivel de significación de 0.05 para la categoría de composición/agregación. Por lo tanto, podemos rechazar la hipótesis nula y en este caso. Esto quiere decir que se puede afirmar, con un riesgo de equivocarse

igual a 5%, que UML₂FP produjo mejores resultados que la técnica del Ifpug para los casos de composición/agregación.

5.4.5 EC5-POST: Quinto experimento controlado con estudiantes de posgrado

Este experimento fue realizado con estudiantes de la Maestría en Informática de la PUCP. Participaron veinte personas, pero se tuvieron que descartar la información de cinco de ellas por lo siguiente: dos ya habían participado en un experimento similar y tres eran estudiantes de pregrado de Ing. Informática que estaban tomando la asignatura en la que se aplicó el experimento como parte de sus optativas de la carrera.

El experimento fue una réplica de los dos anteriores (EC4-POST y EC3-PREG) con la diferencia de que se emplearon las situaciones del primer experimento controlado con profesionales EC1-PROF, con la mejora que se identificó al finalizar el experimento con docentes y asistentes de docencia EC2-DOC. Esto quiere decir que las situaciones que se utilizaron para aplicar la técnica del Ifpug en EC4-POST y EC3-PREG se emplearon para UML₂FP en este experimento controlado y viceversa.

Los estudiantes tenían al menos dos años de experiencia en proyectos de software y como parte de la asignatura donde se aplicó el experimento, se había hecho un repaso a las técnicas orientadas a objetos. La Tabla 5-21 muestra el detalle de los resultados obtenidos por cada participante.

Tabla 5-21. Resultados obtenidos por cada participante en EC5-POST

Participante	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
1	1	1	1	1	0	1	0	0
2	2	2	2	2	1	1	0	0
3	2	2	1	1	0	1	0	0
4	2	1	1	1	0	0	1	1
5	1	2	2	2	0	1	0	1
6	2	2	1	2	1	1	1	1
7	2	1	2	1	1	0	1	1
8	2	1	2	2	0	0	0	0
9	2	1	2	2	1	1	1	1
10	2	0	2	2	1	1	0	1
11	2	1	2	2	1	1	1	1
12	0	2	0	0	0	1	0	0
13	2	2	2	2	1	1	1	1
14	2	2	1	2	0	1	0	0
15	2	1	1	0	0	1	1	1

Luego de procesar los datos de la Tabla 5-21, se obtuvo la información estadística que se presenta en la Tabla 5-22.

Tabla 5-22. Valores estadísticos de los resultados obtenidos en EC5-POST

Variable	Asociación		Composición/ Agregación		Clase asociación		Generalización	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
Observaciones	15	15	15	15	15	15	15	15
Mínimo	0	0	0	0	0	0	0	0
Máximo	2	2	2	2	1	1	1	1
Media	1.733	1.4	1.467	1.467	0.467	0.8	0.467	0.6
Desv. Estándar	0.594	0.632	0.64	0.743	0.516	0.414	0.516	0.507

Como se puede observar en la tabla anterior, las medias que se obtuvieron con las dos técnicas son muy parecidas en casi todas las categorías. Para obtener resultados más concluyentes, se tuvieron que aplicar más pruebas estadísticas, las cuales se detallan a continuación.

Se tuvo que aplicar la prueba de Shapiro-Wilk para poder determinar si las muestras de las categorías asociación y de composición/agregación siguen una distribución normal. La Tabla 5-18 muestra los resultados obtenidos con este tipo de prueba. Para las categorías restantes no se aplicó esta prueba, porque los valores que podrían tomar es cero o uno, por consiguiente no siguen una distribución normal.

Tabla 5-23. Resultados de la prueba de Shapiro-Wilk para asociación y composición/agregación en EC5-POST

Variable	Asociación		Composición/ Agregación	
	Ifpug	UML ₂ FP	Ifpug	UML ₂ FP
W	0.525	0.761	0.716	0.643
p-valor	<0.0001	0.001	0.000	<0.0001

Dado que el p-valor calculado fue menor que el nivel de significación establecido ($\alpha = 0.05$) se rechaza la hipótesis de distribución normal para todas las categorías. Debido a los resultados obtenidos, se tuvo que emplear la prueba no paramétrica de los rangos señalados de Wilcoxon, considerando las hipótesis estadísticas definidas al inicio de esta sección.

Tabla 5-24. Resultados de la prueba de los rangos señalados de Wilcoxon en EC5-POST por categoría

Variable	Asociación	Composición / agregación	Clase asociación	Generalización
V	74.500	27.000	12.000	0.000
Valor esperado	49.500	27.000	42.000	14.500
Varianza (V)	280.125	182.250	252.000	105.125
p-valor (dos colas)	0.143	0.515	0.063	0.172

Según los resultados mostrados en la tabla anterior, el p-valor obtenido para todas las categorías fue mayor al nivel de significación $\alpha = 0.05$. Esto quiere decir que se puede afirmar, con un riesgo de equivocarse igual a 5%, que no hay diferencias significativas entre los resultados obtenidos con Ifpug y UML₂FP para todas las categorías. De acuerdo a los resultados obtenidos en los experimentos, se puede afirmar que la técnica UML₂FP provee por lo menos resultados igual de precisos que la técnica del Ifpug para determinar los ficheros y sus RET.

5.4.6 Resultados del cuestionario sobre percepción de las técnicas

Una de las tareas que tuvieron que realizar los participantes de los experimentos fue el completar un cuestionario en el que se les consulta la percepción que tuvieron sobre las dos técnicas empleadas. La Tabla 5-19 muestra la cantidad de cuestionarios que se procesaron por cada experimento. En el caso de EC3-PREG y EC4-POST, hubieron dos cuestionarios, uno por cada experimento, que no fueron completados (cuestionarios faltantes).

Tabla 5-25. Cuestionarios procesados por cada experimento controlado

Experimento controlado	Cuestionarios Procesados	Cuestionario Faltantes
EC1-PROF	16	0
EC3-PREG	21	1
EC4-POST	13	1

La Figura 5-3 presenta los resultados de la pregunta ¿cuál de las dos técnicas le parece que es más clara de aplicar? En la figura, la barra de color más claro muestra las respuestas de los estudiantes de pregrado y las barras más oscuras, de los profesionales y de los estudiantes de posgrado.

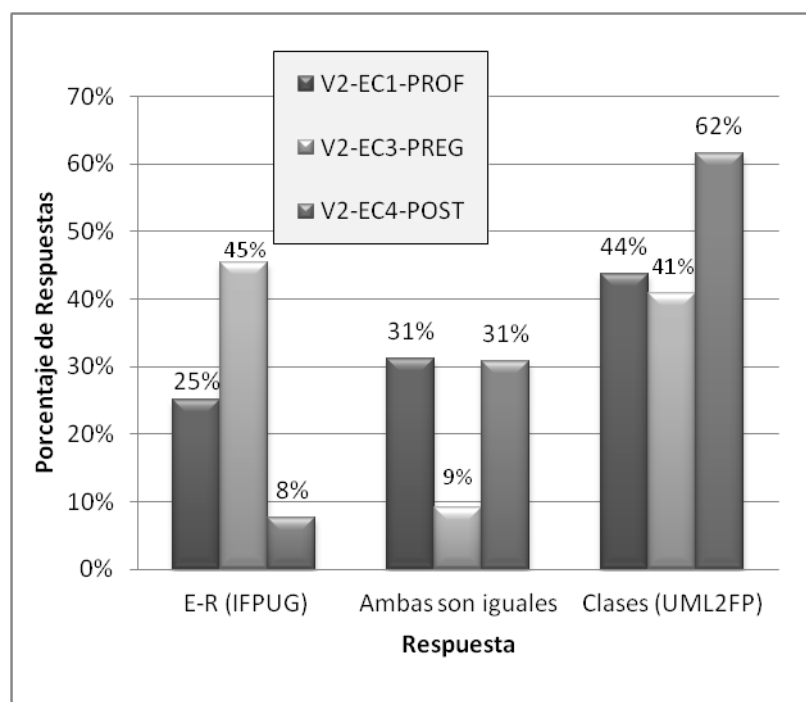


Figura 5-3. Resultados de la pregunta relacionada a cuál de las técnicas es más clara de aplicar

Se puede observar que la mayoría de los estudiantes de posgrado y profesionales piensa que UML₂FP es más clara de aplicar que la técnica del Ifpug. Además, se puede notar que el porcentaje de respuestas de que la técnica del Ifpug es más clara que UML₂FP fue incluso menor al porcentaje de respuestas asociadas a que ambas técnicas son igual de claras.

Los estudiantes de pregrado tuvieron una percepción diferente a los estudiantes de posgrado y profesionales. El porcentaje de participantes que considera que la técnica propuesta por Ifpug es más clara que UML₂FP fue mayor (45% frente a 42%). El porcentaje de alumnos de pregrado que consideraron que ambas técnicas son similares fue mucho menor al considerado por los profesionales y estudiantes de posgrado (9% frente a un promedio de 30% aproximadamente). A pesar de que los estudiantes obtienen mejores resultados con UML₂FP, al consultarles a algunos de ellos, comentaron que UML₂FP tiene más reglas que la técnica del Ifpug y que es más difícil de aplicar.

Adicionalmente, se revisaron los comentarios realizados por los participantes de todos los experimentos. La diferencia entre las opiniones entre los estudiantes de pregrado y los estudiantes de posgrado puede deberse a la interpretación de la pregunta. Parece que los estudiantes pregrado interpretaron la pregunta como "cuál es la técnica más fácil", mientras que los estudiantes de posgrado y profesionales la interpretaron como "cuál es la técnica más eficaz". Debido a estos malentendidos, se planea cambiar este cuestionario en experimentos controlados futuros.

5.5 Amenazas a la Validez de los Experimentos

En esta sección se discuten las diferentes amenazas en la validez de esta evaluación y cómo se intentaron aliviar.

5.5.1 Validez de la construcción

La validez de la construcción es el grado con el cual las variables independientes y dependientes son medidas con precisión por los instrumentos empleados en el estudio.

El registro de los resultados en los formularios permitió medir cuantitativamente la precisión de las técnicas a comparar a través de la correcta identificación de ficheros lógicos cuando los participantes tuvieron que aplicar ambas técnicas. Además, la variable dependiente que fue usada, precisión, es propuesta en ISO/IEC 14143-3 [ISO, 2003].

5.5.2 Validez interna

Mediante el análisis de los resultados del experimento, se puede concluir que existe evidencia empírica que muestra que hay una relación entre las variables independientes y la dependiente. Se ha tratado de aliviar los diferentes aspectos que podrían poner en peligro la validez interna del estudio:

- **Diferencia entre participantes.** En el caso de los profesionales del experimento EC1-PROF y que además no eran alumnos de posgrado de la PUCP, se les hizo una revisión rápida de diagramas E-R, diagramas de clase y una introducción sobre PF para que puedan aplicar las reglas de conversión a ficheros de Ífpug y UML₂FP. Los estudiantes de pregrado y posgrado que participaron en los experimentos EC3-PREG, EC4-POST y EC5-POST habían tomado clases sobre diagramas de clases de análisis y todos conocían sobre diagramas E-R. También, antes de la ejecución de los experimentos, se les había presentado una breve introducción de la técnica PF.
- **Efectos de aprendizaje.** La aplicación de casos o situaciones diferentes canceló el efecto de aprendizaje, efecto que podría haberse producido si se hubieran aplicado las mismas situaciones o casos con ambas técnicas.
- **Conocimiento.** Se emplearon los mismos casos para todos los sujetos.
- **Efectos de fatiga.** A cada participante le tomó aproximadamente 50 minutos en promedio para aplicar ambos casos de estudio y responder los cuestionarios. Por lo tanto, la fatiga no fue relevante.
- **Efectos de persistencia.** La mayoría de participantes no habían hecho un experimento similar anteriormente. Se descartaron los datos de aquellos participantes que habían estado en un experimento controlado similar a los ejecutados.

- **Motivación de los participantes.** Los estudiantes de pregrado y posgrado estuvieron motivados, porque tenían que aplicar la técnica de PF para los trabajos que se les asignaron para el curso en el que estaban matriculados. En el caso de los profesionales que participaron en EC1-PROF, estuvieron motivados porque querían conocer sobre nuevas técnicas para mejorar su trabajo y algunos de ellos estaban trabajando en compañías que estaban mejorando sus procesos de construcción de software.

5.5.3 Validez externa

Se identificaron dos amenazas a validez externa que limita la capacidad para aplicar cualquier generalización de nuestras conclusiones, las cuales son similares a la evaluación de la primera versión a conversión a ficheros:

- **Materiales.** Se trató de emplear casos representativos a casos de sistemas de información reales, en los cuales los estudiantes tuvieron que identificar relaciones de asociación, agregación y composición entre clases. Sin embargo, se necesitarían realizar más estudios en el que se empleen casos o situaciones reales.
- **Sujetos.** Aunque los estudiantes de posgrado y profesionales tenían experiencia en proyectos de desarrollo de software en sus respectivos trabajos, no habían empleado técnicas conocidas para la estimación en proyectos de software, por lo que se les podría considerar como inexpertos en estimación, al igual que los alumnos de pregrado. Se necesitaría realizar más experimentos con profesionales que tengan experiencia en técnicas de estimación para poder generalizar estos resultados.

5.6 Conclusiones de la Evaluación de las Reglas de Conversión a Ficheros de UML₂FP

La Tabla 5-26 muestra un resumen de los resultados obtenidos en cada experimento controlado realizado para evaluar las reglas de conversión a ficheros. La primera columna de la tabla muestra el nombre del experimento, la segunda a la quinta presenta con qué técnica se obtuvo mejores resultados por cada categoría y la última indica los tipos de situaciones que se emplearon para aplicar las técnicas.

Tabla 5-26. Resultados obtenidos en cada experimento para evaluación de las reglas de conversión a ficheros

Experimento Controlado	Mejor Técnica por Categoría				Situaciones o casos empleados
	Asociación	Composición/Agregación	Clase asociación	Generalización	
EC1-PROF	Ifpug	Igual	Igual	Igual	Inicial
EP2-DOC	UML ₂ FP	UML ₂ FP	Igual	Igual	Invertido al inicial
EC3-PREG	Igual	UML ₂ FP	Igual	Igual	Invertido al inicial mejorado
EC4-POST	Igual	UML ₂ FP	Igual	Igual	Invertido al inicial mejorado
EC5-POST	Igual	Igual	Igual	Igual	Inicial mejorado

Como se puede observar en la tabla anterior, en la categoría de asociación, se obtuvieron mejores resultados con Ifpug en el primer experimento controlado EC1-PROF y se esperaba que se obtuvieran los mismos resultados, ya que ambas técnicas proponen reglas equivalentes. Debido a estos resultados, se realizó un experimento piloto con docentes y asistentes de docencia para verificar si la descripción de las situaciones era correcta. Para ello, se invirtieron las situaciones, de tal manera que los casos empleados en E-R se adaptaron para OO y los de OO se adaptaron para E-R. Se pudo determinar que a una de las situaciones le faltaba una mayor descripción. Con esta modificación, se pudo obtener los mismos resultados para ambas técnicas en los siguientes experimentos controlados.

En cuanto a las categorías de clase asociación y de generalización, en todos los experimentos se obtuvieron los mismos resultados con ambas técnicas. Esto es comprensible, ya que las reglas de UML₂FP son similares a las definidas por el Ifpug.

Para la categoría composición/agregación, se obtuvieron mejores resultados para UML₂FP en dos de los experimentos controlados: uno con estudiantes de posgrado y otro con estudiantes de pregrado. También, se produjeron los mismos resultados en ambas técnicas cuando se emplearon las situaciones iniciales con y sin mejora en los experimentos controlados EC1-PROF y EC5-POST. Esto podría indicar que el orden de las situaciones definidas podría haber afectado el resultado de la aplicación de las técnicas. Sin embargo, el cambio de las situaciones no produjo que se obtuvieran mejores resultados con la técnica del Ifpug. Por lo tanto, se podría afirmar que para la categoría composición/agregación por lo menos se obtienen resultados similares con UML₂FP que con Ifpug.

Según los resultados obtenidos en los experimentos controlados, podemos observar que la precisión que se obtiene al aplicar UML₂FP para identificar ficheros es por lo menos similar que al aplicar la técnica definida por Ifpug. En cuanto a la diferencia de precisión obtenida para las dos técnicas entre estudiantes de pregrado y profesionales (a los estudiantes de posgrado se le

puede considerar como profesionales), se puede decir que los resultados son muy parecidos. Aunque para los estudiantes de pregrado, las reglas de composición/agregación produjeron mejores resultados para UML₂FP, también lo fue para algunos de los experimentos con profesionales.

En cuanto al cuestionario en el que los participantes tuvieron que indicar su percepción sobre las técnicas, hubo una diferencia en las respuestas registradas por lo estudiantes de pregrado y por las indicadas por los estudiantes de posgrado y profesionales. Aunque con los estudiantes de pregrado no se obtuvieron mejores resultados con Ifpug, la mayoría tiene una mejor percepción de ésta que de UML₂FP. A diferencia de los profesionales y de los estudiantes de posgrado, cuya mayoría tiene una mejor percepción de UML₂FP que de Ifpug.

Cabe resaltar que UML₂FP tiene más reglas que Ifpug y al parecer, los estudiantes de pregrado interpretaron la pregunta *"¿Cuál de las dos técnicas le parece que es más clara de aplicar?"* como *"¿Cuál es la técnica más fácil?"*. Mientras que los estudiantes de posgrado y profesionales la interpretaron como *"¿Cuál es la técnica más eficaz?"*. Debido a estos malentendidos, se planea cambiar este cuestionario en futuras evaluaciones con preguntas más precisas.

Evaluación del Diagrama de Precedencias de Casos de Uso

Capítulo

6

Este capítulo tiene como finalidad evaluar el diagrama de precedencias de casos de uso (UCPD) como herramienta para poder definir la secuencia de construcción de casos de uso.

Inicialmente se evaluó la percepción que tuvieron alumnos de pregrado de cuarto año del programa de Ingeniería Informática de la Pontificia Universidad Católica del Perú (PUCP) que utilizaron la técnica como parte de un proyecto en la asignatura de Ingeniería de Software. Aunque los resultados fueron muy alentadores, se diseñaron y ejecutaron experimentos controlados con profesionales a fin de determinar la viabilidad de empleo de UCPD en la industria. Los profesionales tuvieron que probar la efectividad de UCPD frente a las técnicas que usualmente ellos emplean en sus trabajos (técnicas ad hoc). De manera complementaria, se llevó a cabo un experimento controlado con alumnos de pregrado de la PUCP con el fin de comparar los resultados obtenidos con los profesionales.

El capítulo se estructura como sigue: la Sección 6.1 presenta una visión general de la evaluación, la Sección 6.2 muestra la evaluación inicial realizada a alumnos de pregrado, la Sección 6.3 detalla la evaluación mediante experimentos controlados con profesionales y alumnos de pregrado, la Sección 6.4 incluye la evaluación de la percepción de UCPD, la Sección 5.5 muestra la evaluación del Modelo de Adopción de Métodos (MAM) y, finalmente, se presentan las conclusiones de la evaluación de UCPD.

6.1 Visión General de la Evaluación

TUPUY ha sido probada en proyectos con alumnos desde el año 2004. UCPD, como parte de TUPUY, fue empleada también en dichos proyectos. Es por ello que en el semestre académico 2005-2 se preparó y diseñó un cuestionario con la finalidad de determinar la percepción que tuvieron los estudiantes sobre UCPD y así saber si es posible su aplicación en la industria.

Aunque los resultados con alumnos fueron alentadores, no se pudo comprobar la efectividad de UCPD frente a las técnicas propias o técnicas ad hoc debido a que ellos sólo tenían experiencia en proyectos de software como parte de asignaturas de la carrera. Por este motivo es que a finales del 2007 y del 2009 se realizaron experimentos controlados con profesionales que tenían más de 2 años de experiencia en la industria, que permitió cuantitativamente determinar la efectividad de UCPD frente a las técnicas propias (técnicas ad hoc)

de los participantes. Además, se pudo determinar la percepción que los profesionales tenían de UCPD, de tal manera que se pudo hacer una comparación con los resultados obtenidos con los alumnos de pregrado. También, se realizó un experimento controlado con alumnos de pregrado en el año 2010 con el fin de comparar sus resultados con los obtenidos con los profesionales.

Para poder llevar a cabo la evaluación de la percepción de los alumnos de pregrado y profesionales, se empleó como base al Modelo de Adopción de Métodos (MAM), cuyo nombre original en inglés es *Method Adoption Model* y fue propuesto por Moody [Moody, 2001]. El MAM está basado en el *Technology Acceptance Model (TAM)* de Davis [Davis, 1989], Modelo de Aceptación de la Tecnología en castellano, que sirve para explicar y predecir la aceptación de la tecnología de la información por parte del usuario en función de un conjunto de constructores, los cuales fueron adaptados por el MAM para explicar y predecir la adopción de métodos. La Figura 6-1 muestra los constructores que proponen el modelo y la relación existente entre ellos.

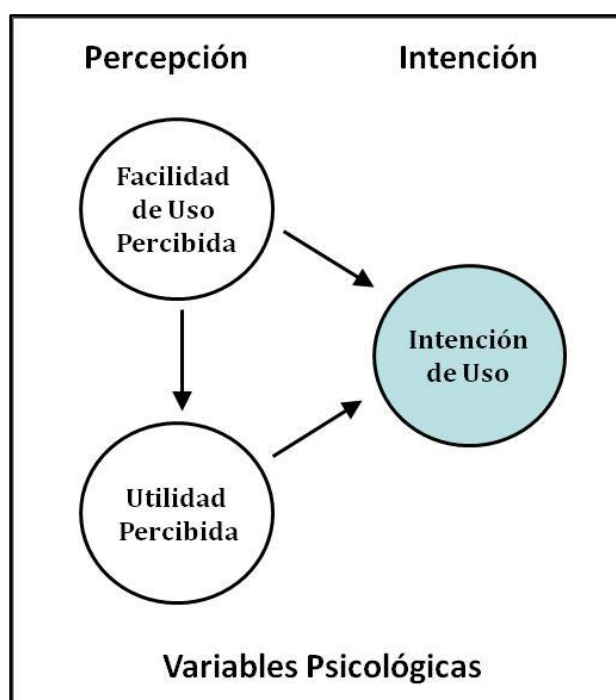


Figura 6-1. MAM propuesto por Moody

Los constructores del MAM se definen a continuación:

- **Facilidad del uso percibida:** Grado en el cual una persona siente que usar un método en particular debería estar libre de esfuerzo.
- **Utilidad percibida:** Grado en el cual una persona siente que un método en particular será efectivo para lograr los objetivos propuestos.

- **Intención de uso:** Grado de intencionalidad que tiene una persona para usar un método particular.

La denominada “adopción en práctica” es determinada por las percepciones de las personas que usan un determinado método. Es por ello que los constructores del MAM son variables de tipo psicológico.

Debido a que se realizaron cuatro estudios (tres experimentos controlados y una evaluación inicial), la Tabla 6-1 muestra lo siguiente: la secuencia de ejecución de los estudios, el tipo de estudio, a quienes se aplicó, el año de realización y la sigla que se empleará en el resto del capítulo para identificarlos.

Tabla 6-1. Tipos de estudios realizados para evaluar UCPD

Secuencia	Tipo de Estudio	Aplicado a	Año de realización	Sigla
1	Evaluación cualitativa	Estudiantes de pregrado	2005	EI-PREG
2	Experimento controlado	Profesionales	2007	EC1-PROF
3	Experimento controlado	Estudiantes de posgrado (profesionales)	2009	EC2-POST
4	Experimento controlado	Estudiantes de pregrado	2010	EC3-PREG

Como se puede observar en la tabla anterior, el primer y último estudio corresponden a estudiantes de pregrado. Para el segundo estudio, se invitaron a profesionales con al menos dos años de experiencia en la industria. En cuanto al tercer estudio, participaron profesionales que, al igual que los participantes del segundo estudio, también tenían por lo menos dos años de experiencia en la industria.

6.2 EI-PREG: Evaluación cualitativa con alumnos de pregrado

El primer estudio que se realizó consistió en una evaluación cualitativa en el que participaron con alumnos de pregrado del cuarto año del programa de Ingeniería Informática de la Pontificia Universidad Católica del Perú. Estos

alumnos emplearon TUPUY, y por ende UCPD, en un proyecto de software. La nota que se asignaría al proyecto formaba parte de la calificación de la asignatura de Ingeniería de Software.

Para realizar esta evaluación se diseñó y aplicó un cuestionario en el que los alumnos tuvieron que opinar sobre los tres constructores del MAM para UCPD. Utilizando el criterio "PICOC" propuesto por Petticrew y Roberts [Petticrew, 2006] y que fue empleado previamente en la revisión sistemática en el estado de la cuestión (ver Capítulo 2), esta evaluación se definió de la siguiente manera:

Población (*population*): desarrollo de software basado en casos de uso

Intervención (*intervention*): la técnica UCPD

Comparación (*comparison*): técnicas ad hoc o propias

Salidas (*outcomes*): percepción positiva de UCPD, según las variables definidas en el MAM.

Contexto (*context*): estudiantes de pregrado del cuarto año de Ingeniería Informática de la Pontificia Universidad Católica del Perú, quienes han empleado la técnica en un proyecto de la asignatura de Ingeniería de Software.

Las preguntas de investigación que se han identificado para esta evaluación son las siguientes:

- ¿UCPD es percibida como fácil de usar?
- ¿UCPD es percibida como útil?
- ¿Existe la intención de usar UCPD en futuros proyectos?

Cabe resaltar que existen evaluaciones similares a la propuesta para determinar si una técnica podría ser utilizada en la práctica. Algunas de ellas se incluyen en las tesis doctorales de Strasunskas [Strasunskas, 2006], Condori [Condori, 2006] y Abrahao [Abrahao, 2004].

A partir de las variables psicológicas “facilidad de uso percibida”, “utilidad percibida” e “intención de uso” incluidas en el MAM, se estudiarán los resultados que se obtengan con alumnos de pregrado con el objetivo de definir si se debería realizar un estudio con profesionales con experiencia para determinar la posible aplicación de UCPD en un entorno profesional.

6.2.1 Sujetos

En este estudio participaron 31 estudiantes de pregrado del cuarto año de la carrera de Ingeniería Informática de la PUCP del semestre 2005-2. Los alumnos utilizaron UCPD en un proyecto del curso de Ingeniería de Software en

ese mismo semestre. El cuestionario fue aplicado al finalizar el proyecto, es decir al finalizar el semestre académico.

6.2.2 Materiales

Los estudiantes tuvieron que responder un cuestionario que incluía diferentes preguntas asociadas a su percepción sobre UCPD. En esta subsección, se han incluido partes del cuestionario aplicado y una explicación de sus preguntas. El anexo A contiene el cuestionario completo que fue aplicado.

Cabe resaltar que a los alumnos que respondieron el cuestionario no se les indicó en ningún momento que la técnica era una propuesta ni quien la proponía con el fin de no influenciarlos en sus respuestas.

En una de las preguntas, los estudiantes tuvieron que indicar cuál es la secuencia de construcción de caso de uso que seguiría si no tuviera restricciones por parte del usuario. La pregunta que se incluyó en el cuestionario se muestra en la Figura 6-2.

Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones	
a)Maestros-transacciones-reportes	b)Maestros-reportes-transacciones
c)Reportes-maestros-transacciones	d)Reportes-transacciones-maestros
e)Transacciones-reportes-maestros	f)Transacciones-maestros-reportes

Figura 6-2. Pregunta sobre secuencia de construcción aplicada en EI-PREG

También, en el mismo cuestionario, se colocaron preguntas relacionadas con la percepción que los mismos alumnos tuvieron de la técnica según los constructores del MAM. Cada respuesta tuvo que ser cuantificada en una escala de cinco niveles de Likert [Likert, 1931]. Como se puede observar en la Figura 6-3, en las preguntas a, b, c, d y e; se colocaron afirmaciones opuestas en cada extremo (izquierda y derecha de la hoja) y los estudiantes tuvieron que marcar de los cinco círculos el estuviera de acuerdo a su opinión. También, se colocaron algunas preguntas en formato opuesto para evitar las respuestas monótonas en los participantes.

5. Marque con una "X" el círculo que esté más cercano a su opinión.

a. Me parece difícil y complejo construir el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Me parece fácil y sencillo construir el diagrama de precedencias
b. Creo que el diagrama de precedencias NO es útil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Creo que el diagrama de precedencias es muy útil para planificar las iteraciones de construcción
c. En general, me parece útil el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	En general, me parece que el diagrama de precedencias NO es útil
d. Me hubiese parecido fácil construir los casos de uso sin seguir la secuencia sugerida por el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Me hubiese parecido difícil construir los casos de uso sin seguir la secuencia sugerida por el diagrama de precedencias
e. Usaré el diagrama de precedencias en próximos proyectos de desarrollo de software	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Definitivamente no usaré el diagrama de precedencias en próximos proyectos de desarrollo de software

Figura 6-3. Preguntas asociadas al MAM empleadas en el cuestionario del estudio EI-PREG

El diseño del cuestionario asociado a la pregunta 5 que se muestra en la Figura 6-3 es una adaptación de los cuestionarios empleados por Condori [Condori, 2006] y Abrahao [Abrahao, 2004] en sus estudios.

La Tabla 6-2 muestran las preguntas que se incluyeron en el cuestionario y al constructor del MAM que corresponden. La pregunta "d" no pertenece a ninguno de los constructores, porque está asociada a la pregunta sobre secuencia de construcción que se muestra en la Figura 6-2.

Tabla 6-2. Preguntas por cada tipo de constructor del MAM para el estudio EI-PREG

Constructor del MAM	Pregunta
Facilidad de uso	a. Me parece difícil y complejo construir el diagrama de precedencias
Utilidad	b. Creo que el diagrama de precedencias NO es útil c. En general, me parece útil el diagrama de precedencias
Intención de uso	e. Usaré el diagrama de precedencias en próximos proyectos de desarrollo de software.

6.2.3 Resultados

En cuanto a la pregunta sobre qué secuencia de construcción de caso de uso seguiría si no tuviera restricciones de usuario, todos los alumnos respondieron que la secuencia que emplearían es *maestros-transacciones-reportes*.

La Tabla 6-3 muestra las respuestas obtenidas sobre las variables del MAM del cuestionario: facilidad de uso percibida (FUP), utilidad percibida (UP) e

intención de uso (IU). La primera columna de la tabla muestra el número de observación y las cuatro columnas restantes, las respuesta por cada pregunta asociada al MAM en un valor según la escala de Likert (rango del 1 al 5).

Tabla 6-3. Respuestas obtenidas sobre las variables del MAM para EI-PREG

Observación	Pregunta			
	5.a (FUP)	5.b (UP)	5.c (UP)	5.e (IU)
1	4	5	5	4
2	3	5	5	4
3	3	5	5	4
4	4	4	4	4
5	5	5	5	5
6	4	3	3	5
7	4	5	5	5
8	3	5	5	5
9	5	5	5	5
10	5	5	5	4
11	3	4	4	4
12	3	5	5	4
13	3	5	5	4
14	5	5	5	5
15	4	4	4	4
16	5	5	5	5

Observación	Pregunta			
	5.b (UP)	5.b (UP)	5.b (UP)	5.b (UP)
17	4	5	4	5
18	4	4	4	4
19	3	4	4	3
20	4	5	3	5
21	5	3	3	3
22	4	5	2	4
23	4	5	5	5
24	5	5	5	5
25	4	4	4	4
26	3	4	4	5
27	4	5	3	2
28	5	4	5	4
29	3	2	3	3
30	4	5	5	5
31	4	3	3	3

Para responder a las preguntas de investigación que se formularon previamente, se determinó que la media de las respuestas de los alumnos debería ser mejor al puntaje medio de la escala de Likert (puntaje igual a 3). Una calificación de tres significa que la percepción es neutral y si es mayor es porque percibe una ventaja al aplicar UCPD. Por ello, la hipótesis estadística para evaluar cada pregunta se definió de la siguiente manera:

$$H_0: \mu \leq 3, \alpha = 0.05$$

$$H_a: \mu > 3$$

Donde α representa el 5% de significación, esto quiere decir que existe un 0.05 de probabilidad de aceptar la hipótesis alternativa cuando la hipótesis nula es verdadera (error de tipo I).

Para el análisis estadístico de la evaluación inicial con alumnos de pregrado, se empleó la herramienta SPSS. En las siguientes subsecciones se muestran los resultados obtenidos para las preguntas sobre facilidad de uso, utilidad e intención de uso.

6.2.3.1 Facilidad de uso percibida

Según los resultados mostrados en la Tabla 6-3 correspondiente a la pregunta sobre facilidad de uso (pregunta 5.a), la Tabla 6-4 presenta el resultado del análisis estadístico descriptivo realizado.

Tabla 6-4. Análisis estadístico descriptivo sobre la facilidad de uso percibida para EI-PREG

Variable	Resultado
Observaciones	31
Mínimo	3
Máximo	5
Media	3.968
Desviación Estándar	0.752

Se puede observar que la media es de 3.968 la cual es superior al puntaje medio (3). El rango de valores osciló entre 3 y 5, por lo que se puede observar que ningún alumno registró un valor menor al puntaje medio (3).

Como los datos de la muestra obtenida son de tipo ordinal, no se pudo aplicar una prueba paramétrica. Por ello, se tuvo que emplear la prueba no paramétrica de los rangos señalados de Wilcoxon, pero como esta prueba emplea la mediana y no la media, se considerarán las hipótesis estadísticas definidas cambiando la media (μ) por mediana. Con esta prueba se obtuvo un estadístico de contraste (W) igual a 253. Puesto que el p-valor para la prueba es menor que 0.05 (p-valor<0.001), puede rechazarse la hipótesis nula. Por lo tanto, se puede afirmar, con un riesgo de equivocarse igual a 5%, que UCPD fue percibida como fácil de usar.

6.2.3.2 Utilidad percibida

La Tabla 6-5 muestra las respuestas obtenidas en las preguntas sobre utilidad percibida: preguntas 5b y 5c del cuestionario. La primera columna de la tabla muestra el número de observación; la segunda columna, las respuestas de las preguntas de utilidad en un valor según la escala de Likert (rango del 1 al 5) y la última columna, el promedio de las respuestas. Dado que se incluyeron dos preguntas para esta variable, para realizar las pruebas estadística, se tomó el promedio redondeado sin decimales de los valores obtenidos.

La pregunta 5b es la misma que la pregunta 5c, pero en sentido inverso. Como se puede observar en la Tabla 6-5, las respuestas obtenidas son muy similares, por lo que no fue necesario descartar ninguna.

Tabla 6-5. Respuestas obtenidas para las preguntas de utilidad para EI-PREG

Observación	Pregunta		Promedio de las Respuestas
	5.b	5.d	
1	5	5	5
2	5	5	5
3	5	5	5
4	4	4	4
5	5	5	5
6	3	3	3
7	5	5	5
8	5	5	5
9	5	5	5
10	5	5	5
11	4	4	4
12	5	5	5
13	5	5	5
14	5	5	5
15	4	4	4
16	5	5	5

Observación	Pregunta		Promedio de las Respuestas
	5.b	5.d	
17	5	4	5
18	4	4	4
19	4	4	4
20	5	3	4
21	3	3	3
22	5	2	4
23	5	5	5
24	5	5	5
25	4	4	4
26	4	4	4
27	5	3	4
28	4	5	5
29	2	3	3
30	5	5	5
31	3	3	3

Dado que para la utilidad percibida se incluyeron dos preguntas, se tuvo que realizar un análisis de fiabilidad. La fiabilidad describe la consistencia en medir el mismo fenómeno en un cierto tiempo por diversas personas y para determinarla se emplea el alfa de Cronbach. Según Nunally [Nunally, 1978] un alfa mayor o igual que 0.7 indica que un constructor es fiable. Para la utilidad percibida, el alfa de Cronbach fue de 0.725; por consiguiente, se puede decir que los resultados obtenidos sobre la utilidad percibida son fiables.

La Tabla 6-6 presenta el resultado del análisis estadístico descriptivo realizado para los datos obtenidos de las preguntas sobre utilidad percibida. Se puede observar que la media es de 4.419 la cual es superior a 3 que es el puntaje medio. El rango de valores osciló entre 3 y 5.

Tabla 6-6. Análisis estadístico descriptivo sobre la utilidad percibida para EI-PREG

Variable	Resultado
Observaciones	31
Mínimo	3
Máximo	5
Media	4.419
Desviación Estándar	0.720

Para determinar si los resultados obtenidos sobre la utilidad percibida siguen una distribución normal, se empleó la prueba de Shapiro-Wilk con un nivel de significación $\alpha = 0.05$. Como el p-valor obtenido con esta prueba fue igual a 0.000, se puede rechazar la hipótesis de distribución normal y se tuvo que aplicar la prueba no paramétrica de los rangos señalados de Wilcoxon.

Al emplear la prueba de los rangos señalados de Wilcoxon para una muestra, se obtuvo un estadístico de contraste (W) igual a 378.0. Dado que el p-valor para la prueba es menor que 0.05 (p-valor < 0.001), se puede rechazar la hipótesis nula (para las hipótesis estadísticas se empleó la mediana y no la media). Por lo tanto, podemos afirmar, con un riesgo de equivocarse igual a 5%, que UCPD fue percibida como útil.

6.2.3.3 Intención de uso

La Tabla 6-7 muestra el resultado del análisis descriptivo realizado para los datos obtenidos de las preguntas sobre intención de uso (ver Tabla 6-3). Se puede observar que la media es de 4.226 la cual es muy superior al puntaje medio (3). El rango de valores osciló entre 2 y 5.

Tabla 6-7. Análisis estadístico descriptivo sobre la intención de uso para EI-PREG

Variable	Resultado
Observaciones	31
Mínimo	2
Máximo	5
Media	4.226
Desviación Estándar	0.805

Debido a que los datos correspondientes a intención de uso son de tipo ordinal, se tuvo que aplicar la prueba no paramétrica de los rangos señalados de Wilcoxon. El estadístico de contraste (W) obtenido con esta prueba fue igual a 370.5. El p-valor obtenido fue menor que 0.05 (p-valor = 0.001), por lo que se puede rechazar la hipótesis nula (para las hipótesis estadísticas se empleó la mediana y no la media). Por lo tanto, podemos afirmar, con un riesgo de equivocarse igual a 5%, que existe una intención para usar UCPD.

6.2.4 Discusión de los resultados

Los resultados obtenidos muestran de manera general una actitud positiva de los alumnos hacia la técnica. Los participantes percibieron que UCPD es fácil de usar y útil, y que también la emplearían en futuros proyectos de software.

Se podría considerar como inconveniente el haber utilizado una pregunta por constructor (excepto el constructor de utilidad percibida), pero existen estudios en los que se ha aplicado este mismo criterio en campos como la medicina con muy buenos resultados y sin que el estudio se invalide por esta causa [Davey, 2007] [Cepeda, 2008] [Temel, 2006]. El objetivo fue crear un cuestionario que sea sencillo y fácil de completar.

Aunque se conoce la ventaja de realizar investigaciones con estudiantes [Carver, 2003] existen amenazas que podrían hacer difícil generalizar los resultados que se obtuvieron a la práctica o la industria. Una de estas amenazas es que los alumnos tienen poca experiencia en proyectos de software y por lo tanto no conocen otras técnicas similares a UCPD. Es por ello que podrían haber calificado a la técnica propuesta con buen puntaje porque es la única que conocen.

Por esta razón es que se decidió realizar una familia de experimentos controlados con profesionales con experiencia en la industria y con alumnos de pregrado. Los detalles de los experimentos se incluyen en la siguiente sección.

6.3 Evaluación mediante experimentos controlados

Para esta evaluación, se empleó una familia de experimentos controlados. En el primer experimento participaron profesionales con al menos dos años de experiencia. Después, teniendo en cuenta la experiencia obtenida en el primer experimento, se realizó una réplica del experimento con estudiantes de posgrado de la Maestría en Informática de la PUCP, pero con algunos cambios. Finalmente, el mismo tipo de experimento controlado utilizado con los estudiantes de posgrado fue aplicado a alumnos de pregrado.

Para el diseño de esta familia de experimentos se consideraron las sugerencias de Juristo y Moreno [Juristo, 2001]. El objetivo de esta familia de experimentos controlados fue comprobar de manera empírica que la efectividad de UCPD es mayor a las técnicas ad hoc empleadas para determinar la secuencia de construcción de casos de uso.

Para definir los experimentos controlados, se ha empleado la palabra “efectividad” (*effectiveness*, en inglés). Según el diccionario Web de Cambridge⁶, se define a la “eficacia” (*efficacy*, en inglés) como “*an ability, especially of a medicine or a method of achieving something, to produce the intended result; effectiveness*”. Esto quiere decir, que eficacia corresponde a una habilidad, especialmente de una medicina o método en lograr algo, para producir el resultado esperado; y también es sinónimo de “efectividad”. Lo que se quería, al realizar los experimentos controlados, era probar que la técnica UCPD provee mejores resultados que las técnicas ad hoc; por lo tanto, el término más adecuado para definir estos experimentos sería la palabra “eficacia”. Sin

⁶ <http://dictionary.cambridge.org>

embargo, el término “eficiencia” (*efficiency*, en inglés) en el diccionario de la Real Academia de la Lengua Española (RAE)⁷ muestra ambigüedad y se podría interpretar como “eficacia”, a pesar que en los diccionarios en inglés, las definiciones de estos dos términos son marcadamente diferentes. En cuanto al término “efectividad”, ambos diccionarios (RAE y Cambridge) concuerdan en su definición. Por ello, se prefirió utilizar el término “efectividad” y no “eficacia”, porque este último podría ser interpretado en castellano como “eficiencia”.

Utilizando el criterio "PICOC" propuesto por Petticrew y Roberts [Petticrew, 2006], los experimentos controlados se definieron de la siguiente manera:

Población (*population*): desarrollo de software basado en casos de uso

Intervención (*intervention*): la técnica UCPD

Comparación (*comparison*): técnicas ad hoc o propias

Salidas (*outcomes*): mayor efectividad de UCPD para definir la secuencia de construcción de casos de uso, considerando que el desarrollador es libre de seleccionar la secuencia de construcción de casos de uso (no existen restricciones por parte del usuario).

Contexto (*context*): profesionales con al menos 2 años de experiencia en proyectos de desarrollo de software. También, de estudiantes de pregrado de cuarto año de Ingeniería Informática de la PUCP.

La pregunta de investigación es ¿la efectividad de UCPD es mayor que las técnicas propias o informales empleadas (técnicas ad hoc)?

En cuanto a la selección de variables, la variable independiente es la técnica empleada por los sujetos de experimentación para definir la secuencia de construcción de casos de uso. La variable dependiente es la efectividad; es decir, el grado de acercamiento entre los resultados medidos y el valor verdadero.

Para este experimento se considera “valor verdadero” al hecho que la secuencia más fácil para construir casos de uso es “*maestros-transacciones-reportes*”, secuencia que fue seleccionada por profesionales consultados previamente a este experimento para contrastar el resultado y que, curiosamente, coincide con la de todos los alumnos de pregrado en el estudio realizado previamente (ver sección 6.2).

Las herramientas que se emplearon para el análisis estadístico de los experimentos controlados incluidos en esta sección son SPSS y XLStat 2009. La primera de ellas fue utilizada para crear los diagramas de cajas y la segunda, para evaluar las hipótesis estadísticas que se definieron.

⁷ <http://www.rae.es/>

6.3.1 EC1-PROF: Primer experimento controlado realizado con profesionales

En este experimento participaron 34 profesionales con al menos 2 años de experiencia en proyectos de desarrollo de software. Muchos de ellos trabajaban en compañías que están mejorando sus procesos de desarrollo de software a fin de obtener el nivel 3 de certificación en CMMI [SEI, 2009] u obtuvieron ya esta certificación.

Antes del experimento, se realizó una clase de entrenamiento de casos de uso con el fin de nivelar el conocimiento de todos los participantes en este tema. Esta clase se realizó cuatro días antes de la fecha del experimento.

6.3.1.1 Materiales y casos de estudio

Los materiales empleados en el experimento fueron dos casos de estudio y cuatro cuestionarios. El primer caso de estudio corresponde a la elaboración de un sistema para un restaurante y el segundo, a un sistema de matrícula para un colegio de educación secundaria. Ambos casos de estudio son del tipo sistema de información. En el anexo D de este documento, se han incluido todos los documentos que se utilizaron en este experimento controlado.

Cada caso de estudio contaba con la siguiente información: diagrama de casos de uso, descripción de cada caso de uso con sus precondiciones e información que se utiliza en cada caso de uso en términos de clases o entidades. La Figura 6-4 muestra como ejemplo uno de los casos de uso incluidos en el caso de estudio de ventas para un restaurante (caso de estudio 1).

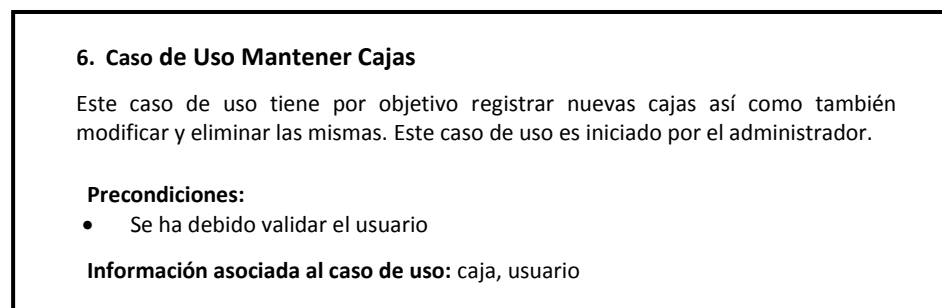


Figura 6-4. Ejemplo de caso de uso incluido en los casos de estudio EC1-PROF

Los cuestionarios 1, 2 y 3 corresponden a preguntas en las que los participantes tenían que decidir entre dos casos de uso y responder cuál de ellos debería ser construido primero. Por ejemplo, para el primer caso de estudio, una de las preguntas fue la siguiente:

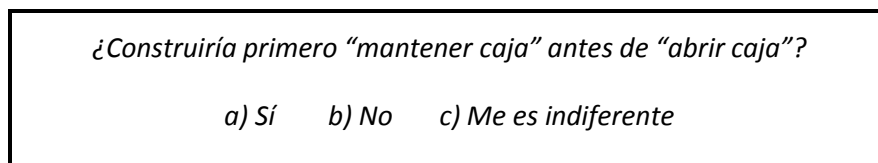


Figura 6-5. Pregunta ejemplo del primer caso de estudio EC1-PROF

Las preguntas se diseñaron de tal manera que los participantes tuvieran que seleccionar entre los siguientes pares de tipos de casos de uso: maestros-transacciones, transacciones-transacciones, maestros-reportes y transacciones-reportes.

El objetivo del cuarto cuestionario fue conocer la percepción de los participantes sobre la facilidad y utilidad de UCPD (las preguntas sobre este tipo son equivalentes a las incluidas en el estudio con los alumnos de pregrado EI-PREG). También, se incluyó una pregunta en la que tuvieron que seleccionar la forma más fácil para construir el software entre casos de uso de tipo maestro, transacción o reporte.

6.3.1.2 Tareas realizadas durante el experimento

Se seleccionó el diseño de experimento intrasujeto; es decir que los profesionales tuvieron que aplicar el primer caso de estudio con las técnicas ad hoc o técnicas propias y en el segundo, con UCPD. Se emplearon dos casos de estudio con similares características (ambos son sistemas de información) a fin de mitigar los efectos de aprendizaje. La Figura 6-6 muestra el diseño del experimento.

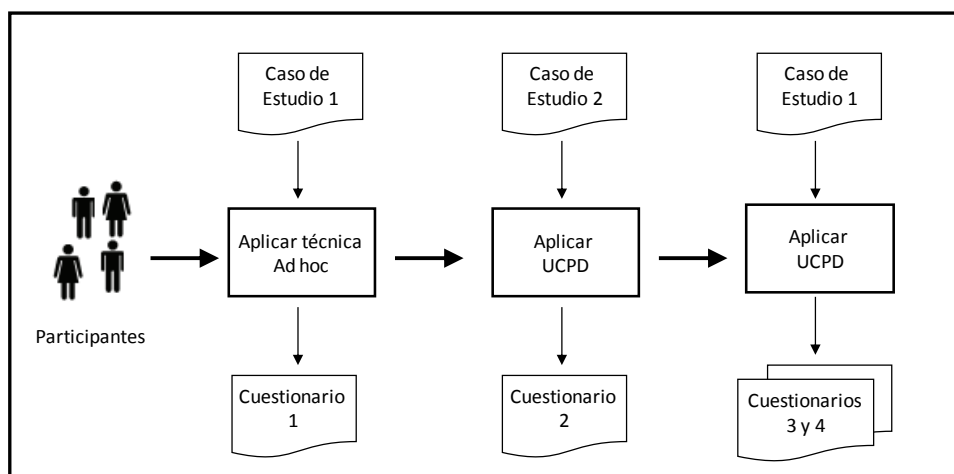


Figura 6-6. Diseño del estudio EC1-PROF

La sesión duró aproximadamente una hora y los profesionales emplearon 45 minutos en promedio para completar todas las tareas. Aunque no fue parte de este estudio conocer que técnica demandó menor tiempo, se pudo observar que los participantes elaboraron cada UCPD en aproximadamente 10 minutos. La Tabla 6-8 presenta las tareas realizadas durante la sesión por los participantes.

Tabla 6-8. Tareas realizadas por los profesionales en EC1-PROF

Tarea Nº	Descripción
1	Recibir caso de estudio 1 y cuestionario 1
2	Completar cuestionario 1
3	Recibir caso de estudio 2 y cuestionario 2
4	Elaborar UCPD para caso de estudio 2
5	Completar cuestionario 2
6	Elaborar UCPD para caso de estudio 1
7	Completar cuestionario 3
8	Completar cuestionario 4

6.3.1.3 Resultados cuantitativos

Para comparar los resultados entre técnicas ad hoc y UCPD, se empleó el porcentaje de “respuestas correctas”, que son aquellas respuestas en las que se considera que la secuencia más fácil de construcción de la funcionalidad de un sistema es *maestros-transacciones-reportes*. Por ejemplo, para el primer caso de estudio que corresponde a un sistema de información para un restaurante, se tienen los siguientes casos de uso con sus descripciones:

- **Mantener mesa:** Este caso de uso tiene por objetivo registrar nuevas mesas así como también modificar y eliminar las mismas. El caso de uso es iniciado por el Jefe de Ventas.
- **Mantener nota de pedido:** Este caso de uso tiene por objetivo registrar nuevas notas de pedido así como también la modificación o eliminación de la información de las notas de pedido ya existentes, mientras aún no se haya generado un documento de pago. El caso de uso es iniciado por el mesero.

El caso de uso Mantener mesa se clasificaría como un caso de uso de mantenimiento de maestros y Mantener nota de pedido, como transacción. El UCPD para estos casos de uso se muestra en la figura siguiente (Figura 6-7):

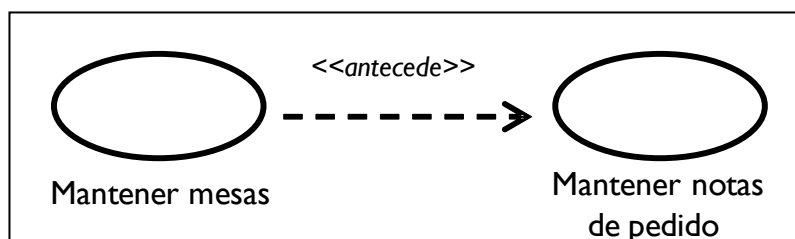


Figura 6-7. Ejemplo de diagrama de precedencias con dos casos de uso

Para estos dos casos de uso aplicaría la segunda regla para definir un UCPD: “Un caso de uso U1 precede a otro caso de uso U2 si U2 necesita información que es registrada por U1” (ver capítulo 4), ya que para poder mantener notas de pedido se tiene que haber registrado previamente la información de las mesas.

Según el UCPD mostrado en la Figura 6-7, para la pregunta del cuestionario 1 “¿Construiría primero ‘Mantener mesas’ antes de ‘Mantener notas de pedido’?” la respuesta sería “Sí”, ya que Mantener mesas se encuentra al lado izquierdo de Mantener notas de pedido y según UCPD la secuencia de construcción debería ser de izquierda a derecha. Cabe resaltar que la respuesta a esa pregunta también está de acuerdo a lo que la mayoría de desarrolladores considera cuál es la secuencia que se debe seguir para construir de manera más sencilla la funcionalidad de un sistema: maestros-transacciones-reportes.

Aunque durante el diseño del experimento se asumió que la secuencia más fácil para construir casos de uso era maestros-transacciones-reportes, por los resultados obtenidos con los alumnos de pregrado (ver sección 6.2) y por opiniones vertidas por algunos profesionales previo a este experimento, se incluyó en el cuestionario 4 la pregunta “seleccionar la secuencia de construcción de casos de uso que usted normalmente seguiría si no tuviera restricciones de usuario: mantener maestros, mantener transacciones y reportes”. El propósito fue sólo confirmar los resultados previamente obtenidos a este experimento.

Como se puede observar en la Figura 6-8, no todos los participantes opinaron que la secuencia más fácil para construir software era la que previamente se había asumido (maestros-transacciones-reportes), solo fue el 76%. Por ello, se tuvieron que revisar con detenimiento los comentarios de los participantes y se tuvo que consultar a algunos de ellos el porqué de la selección de otras alternativas luego de finalizado el experimento.

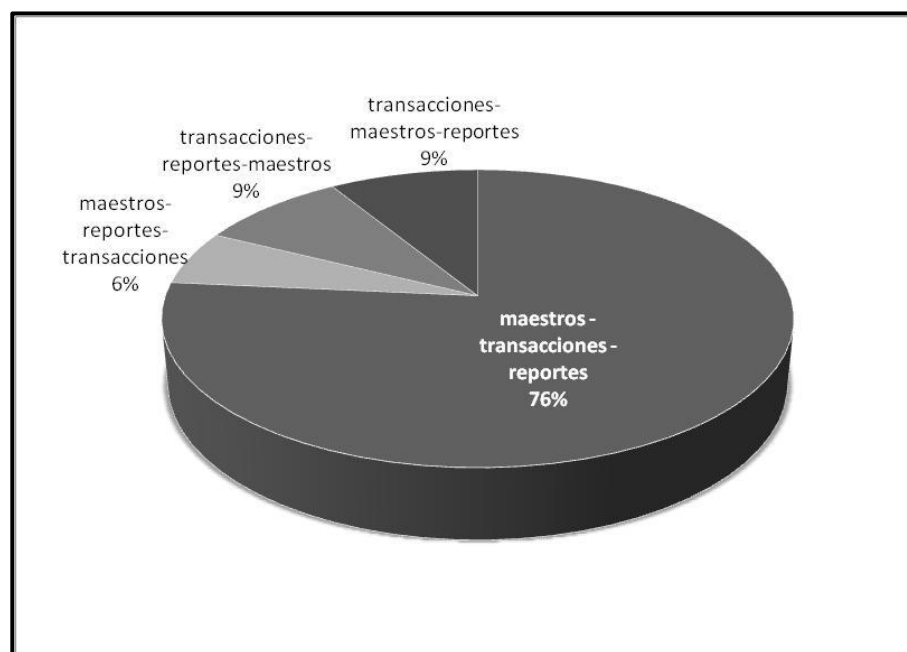


Figura 6-8. Secuencia de construcción seleccionada por los profesionales del estudio EC1-PROF

Algunos consideraron que los “maestros” son fáciles de implementar, por lo que su desarrollo no era difícil o crítico, y en un proceso de construcción deberíamos enfocarnos en las transacciones y en los reportes.

Las secuencias que no seleccionaron fueron reportes-maestros-transacciones y reportes-transacciones-maestros. Debido a los resultados obtenidos en la pregunta sobre la secuencia de construcción (ver Figura 6-8), se analizaron de manera separada los resultados obtenidos con aquellos profesionales que seleccionaron como secuencia a *maestros-transacciones-reportes*, de los que seleccionaron otra secuencia. Se tuvo que descartar los resultados de un participante que no entendió como elaborar el UCPD.

a) Participantes que seleccionaron *maestros-transacciones-reportes*

Fueron veinticinco los participantes de este estudio que seleccionaron a la secuencia de *maestros-transacciones-reportes* como la más fácil para construir software. La Tabla 6-9 y la Tabla 6-10 muestran los resultados obtenidos en los cuestionarios 1 y 2 del estudio para estos participantes. El valor de “1” indica que el participante acertó en la respuesta y “0” que respondió de manera incorrecta. La primera columna muestra el número de participante, la segunda columna hasta la séptima presentan los resultados que obtuvo cada participante en cada pregunta y la última columna muestra el porcentaje de respuestas correctas.

Tabla 6-9. Resultados obtenidos en el cuestionario 1 del estudio EC1-PROF

Participante	Pregunta						Porcentaje de Respuestas Correctas
	1	2	3	4	5	6	
1	0	0	0	0	0	0	0.0%
2	1	0	0	1	0	0	33.3%
3	1	1	1	1	1	1	100.0%
4	1	1	1	1	1	1	100.0%
5	1	1	1	0	1	1	83.3%
6	1	0	1	1	1	1	83.3%
7	1	0	1	1	1	1	83.3%
8	1	0	1	1	1	1	83.3%
9	1	1	0	1	1	1	83.3%
10	1	0	1	1	1	1	83.3%
11	1	0	1	1	1	0	66.7%
12	1	1	0	1	0	0	50.0%
13	1	1	1	1	1	1	100.0%

Participante	Pregunta						Porcentaje de Respuestas Correctas
	1	2	3	4	5	6	
14	1	0	1	0	1	1	66.7%
15	1	1	1	1	1	1	100.0%
16	1	1	1	1	0	1	83.3%
17	1	1	1	1	1	1	100.0%
18	1	1	1	1	1	0	83.3%
19	1	0	1	1	1	1	83.3%
20	1	0	1	1	0	1	66.7%
21	1	1	1	1	1	0	83.3%
22	1	1	1	1	1	1	100.0%
23	1	0	1	1	1	1	83.3%
24	1	0	1	1	1	0	66.7%
25	1	1	1	1	0	0	66.7%

Tabla 6-10. Resultados obtenidos en el cuestionario 2 del estudio EC1-PROF

Participante	Pregunta					Porcentaje de Respuestas Correctas
	1	2	3	4	5	
1	1	1	1	0	1	80.0%
2	1	1	0	1	1	80.0%
3	1	1	0	0	1	60.0%
4	0	1	0	0	1	40.0%
5	1	1	1	1	1	100.0%
6	1	1	1	1	1	100.0%
7	0	1	1	0	1	60.0%
8	1	1	1	1	1	100.0%
9	1	1	1	1	1	100.0%
10	1	1	1	1	1	100.0%
11	1	1	1	1	1	100.0%
12	1	1	1	1	1	100.0%
13	1	1	1	1	1	100.0%

Participante	Pregunta					Porcentaje de Respuestas Correctas
	1	2	3	4	5	
14	1	1	1	1	1	100.0%
15	1	1	1	1	1	100.0%
16	1	1	1	1	1	100.0%
17	1	1	1	1	1	100.0%
18	1	1	1	1	1	100.0%
19	1	1	0	0	1	60.0%
20	1	1	1	1	1	100.0%
21	1	1	1	1	1	100.0%
22	1	1	1	1	1	100.0%
23	1	1	1	1	1	100.0%
24	1	1	1	1	1	100.0%
25	1	0	1	1	1	80.0%

Aunque en los cuestionarios 1 y 2 incluyeron preguntas relacionadas con la secuencia de realización de las pruebas, sólo se han empleado para esta evaluación las respuestas sobre construcción. Los resultados obtenidos sobre la parte de pruebas se emplearán en próximos estudios. La Figura 6-9 muestra el diagrama de cajas con los resultados obtenidos al aplicar las técnicas propias de cada participante (técnicas ad hoc) con el caso de estudio 1 y al aplicar UCPD con el caso de estudio 2.

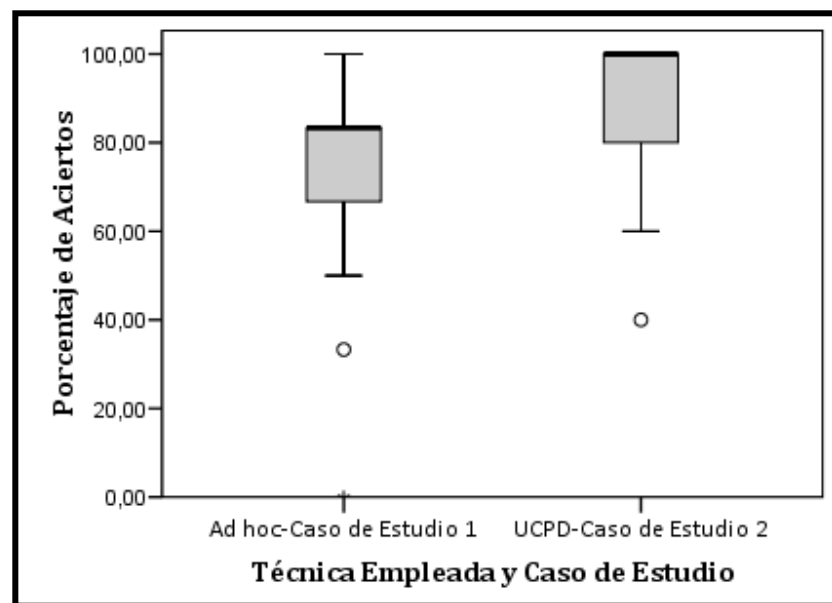


Figura 6-9. Diferencias entre la aplicación de técnicas ad hoc (sin UCPD) y de UCPD con casos de estudio diferentes para EC1-PROF

En un diagrama de cajas, la caja contiene el 50% de los datos, mientras que la línea horizontal en el centro de la caja marca la mediana de la distribución (la mediana es el valor de la variable que deja el mismo número de datos antes y después que él, una vez que se ordenan). Como se puede observar en la Figura 6-9, el valor de la mediana es mayor al aplicar UCPD, porque ésta es de 100% de aciertos, y la diferencia entre los valores mínimos y máximos (valores que se observan con los bigotes de las cajas) es menor también al aplicar UCPD (tomar en cuenta que el bigote superior en UCPD se superpone a la mediana que es de 100%). Además, según el diagrama, el 50% de los datos de la aplicación de UCPD corresponde a la máxima cantidad de aciertos: 100% de aciertos, a diferencia de las técnicas ad hoc, en las que solo el 25% de los datos se encuentran por arriba del 80% de aciertos. Según el diagrama, los resultados obtenidos al utilizar UCPD son mejores que al emplear las técnicas ad hoc.

A pesar de los resultados observados en la Figura 6-9, se realizaron más pruebas estadísticas con la información recopilada en los cuestionarios. La Tabla 6-11 presenta la información estadística descriptiva obtenida luego de

procesar los datos de los cuestionarios que se muestran en la Tabla 6-9 y la Tabla 6-10. La segunda columna muestra los resultados obtenidos con las técnicas que los participantes emplean en su trabajo (técnicas ad hoc) y la tercera, los resultados con UCPD. Para las técnicas ad hoc se empleó el caso de estudio relacionado con un sistema de restaurantes (caso de estudio 1) y para UCPD se utilizó el caso de estudio relacionado con un sistema para un colegio de educación secundaria (caso de estudio 2).

Tabla 6-11. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para el estudio EC1-PROF

Variable	Técnica Ad hoc con Caso de Estudio 1	UCPD con Caso de Estudio 2
Observaciones	25	25
Mínimo	0.0%	40.0%
Máximo	100.00%	100.00%
Media	77.33%	90.40%
Desv. Estándar	23.014	17.436

A fin de determinar si los resultados obtenidos con cada técnica siguen una distribución normal, se aplicó la prueba de Shapiro-Wilk. La Tabla 6-12 muestra los resultados obtenidos con este tipo de prueba.

Tabla 6-12. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para el estudio EC1-PROF

Variable	Técnica Ad hoc con Caso de Estudio 1	UCPD con Caso de Estudio 2
W	0.785	0.616
P-valor	0.000	<0.0001

Dado que el p-valor calculado que se muestra en la tabla anterior fue menor que el nivel de significación establecido ($\alpha = 0.05$) se rechaza la hipótesis de distribución normal para los dos tipos de técnicas. Debido a los resultados obtenidos, no se puede utilizar técnicas paramétricas para probar esos resultados. Es por ello que se utilizó la prueba no paramétrica de los rangos señalados de Wilcoxon.

Las hipótesis estadísticas formuladas para probar la efectividad de ambas técnicas son las siguientes:

$$H_0: \text{Med}_1 \geq \text{Med}_2, \quad \alpha = 0.05$$

$$H_a: \text{Med}_1 < \text{Med}_2$$

Donde Med_1 es la mediana del porcentaje de acierto obtenido con las técnicas ad hoc y Med_2 , con UCPD.

Tabla 6-13. Resultados de la prueba de los rangos señalados de Wilcoxon de técnicas ad hoc vs. UCPD para EC1-PROF

Variable	Resultado
V	76.000
Valor esperado	157.500
Varianza (V)	1357.375
P-valor (una cola)	0.014

Dado que el valor computado es menor que el nivel de significación $\alpha=0.05$, se rechaza la hipótesis nula H_0 . Esto quiere decir que se pudo comprobar empíricamente que UCPD produjo mejores resultados (efectividad mayor) que los obtenidos con las técnicas ad hoc. Esto corrobora las observaciones realizadas con el diagrama de cajas (ver Figura 6-9).

Aunque los resultados previamente presentados son indicativos de los beneficios de la técnica frente a las técnicas ad hoc, se aplicó un tercer cuestionario en el que los profesionales tuvieron que resolver el caso de estudio 1 empleando UCPD. La Tabla 6-14 muestra los resultados obtenidos en el cuestionario 3. El valor de “1” indica que acertó en la respuesta y “0” es respuesta errónea. La última columna muestra el porcentaje de respuestas que acertaron los participantes.

Tabla 6-14. Resultados obtenidos en el cuestionario 3 para el estudio EC1-PROF

Participante	Pregunta						Porcentaje de Respuestas Correctas
	1	2	3	4	5	6	
1	1	1	1	0	1	1	83.3%
2	1	1	1	1	1	1	100.0%
3	1	1	1	1	1	1	100.0%
4	1	1	1	1	0	1	83.3%
5	1	1	0	1	1	1	83.3%
6	1	0	1	1	1	0	66.7%
7	1	0	1	1	1	1	83.3%
8	1	1	0	1	1	1	83.3%
9	1	1	1	1	1	1	100.0%
10	1	1	1	1	0	1	83.3%
11	1	1	1	1	1	0	83.3%
12	1	1	1	1	1	1	100.0%
13	1	1	1	1	1	1	100.0%

Participante	Pregunta						Porcentaje de Respuestas Correctas
	1	2	3	4	5	6	
14	1	1	1	1	0	1	83.3%
15	1	1	0	1	1	1	83.3%
16	1	1	1	1	1	1	100.0%
17	1	1	1	1	1	1	100.0%
18	1	1	1	1	0	1	83.3%
19	1	1	1	1	1	1	100.0%
20	1	1	1	1	0	1	83.3%
21	1	1	1	1	0	1	83.3%
22	1	1	0	1	1	1	83.3%
23	1	1	0	1	1	1	83.3%
24	1	1	1	1	0	1	83.3%
25	1	1	1	1	0	0	66.7%

La Figura 6-10 muestra el diagrama de cajas con los resultados obtenidos al aplicar las técnicas propias de cada participante (técnicas ad hoc) con el caso y al aplicar UCPD, ambos con el caso de estudio 1.

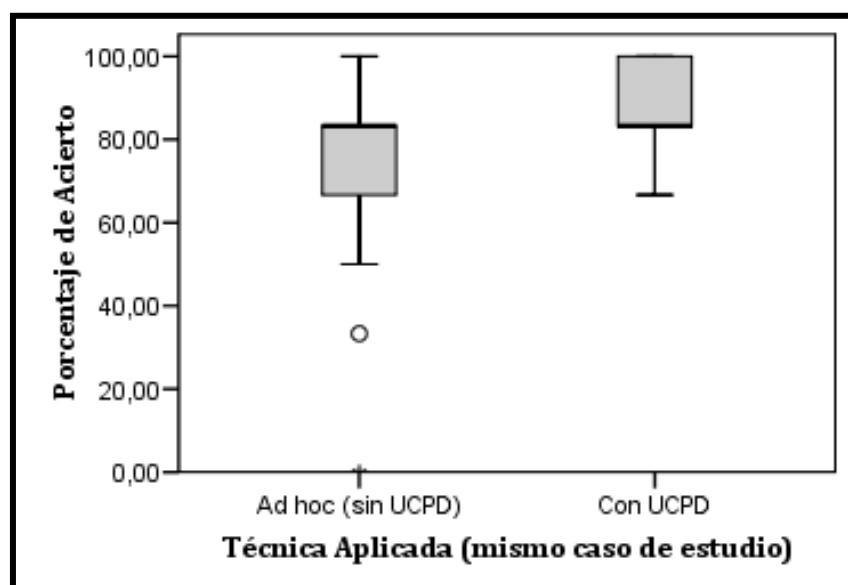


Figura 6-10. Diferencias entre la aplicación de técnicas ad hoc y de UCPD con el mismo caso de estudio para EC1-PROF

Como se puede observar en la Figura 6-10, el valor de las medianas es la misma al aplicar cualquiera de las dos técnicas. Sin embargo, la caja de UCPD se encuentra por encima de la caja de las técnicas ad hoc y la diferencia entre los valores mínimos y máximos (valores que se observan con los bigotes de las cajas) es menor también al aplicar UCPD (tomar en cuenta que el bigote superior en UCPD se superpone a la mediana que es de 100%). Además, el 75% de los datos al aplicar UCPD se encuentran por encima del 80% de aciertos, a diferencia con las técnicas ad hoc en las que sólo el 25% de los datos se encuentran por encima del 80% de aciertos. Según el diagrama, los resultados obtenidos al utilizar UCPD son mejores que al emplear las técnicas ad hoc.

A pesar de los resultados observados en la Figura 6-10, se realizaron más pruebas estadísticas con la información recopilada en los cuestionarios. La Tabla 6-15 presenta la información estadística descriptiva obtenida luego de procesar los datos de los cuestionarios que se muestran en la Tabla 6-9 y la Tabla 6-14. La segunda columna de la tabla muestra los resultados obtenidos al aplicar las técnicas ad hoc y la tercera columna, con UCPD. En ambos casos emplearon el caso de estudio correspondiente a un sistema para un restaurante (caso de estudio 1).

Tabla 6-15. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para el estudio EC1-PROF

Variable	Técnica Ad hoc con Caso de Estudio 1	UCPD con Caso de Estudio 1
Observaciones	25	25
Mínimo	0.0%	66.67%
Máximo	100.00%	100.00%
Media	77.33%	87.33%
Desviación estándar	23.014	9.954

Como se puede observar en la tabla anterior (Tabla 6-15), hay una diferencia significativa entre ambas muestras, ya que la media obtenida con UCPD es mayor que la obtenida con las técnicas ad hoc. Esto podría indicarnos que UCPD dio mejores resultados; sin embargo, para obtener resultados más concluyentes, se tuvieron que aplicar más pruebas estadísticas, las cuales se detallan a continuación.

A fin de determinar si los resultados obtenidos al aplicar UCPD con el caso de estudio 1 siguen una distribución normal, se aplicó la prueba de Shapiro-Wilk. La Tabla 6-16 muestra los resultados obtenidos con este tipo de prueba.

Tabla 6-16. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para EC1-PROF

Variable	UCPD con Caso de Estudio 1
W	0.757
P-valor	<0.0001

Dado que el p-valor calculado que se muestra en la tabla anterior es menor que el nivel de significación $\alpha = 0.05$, se puede rechazar la hipótesis de distribución normal de la muestra. Debido a que las muestras obtenidas al aplicar UCPD en los dos casos de estudio no siguen una distribución normal (ver Tabla 6-12 para la prueba de normalidad para las técnicas ad hoc con caso de estudio 1), no se puede aplicar la prueba *t-student* de las muestras pareadas, por lo que se tiene que seleccionar una prueba no paramétrica. Debido al tipo de datos obtenidos, se emplea la prueba de los rangos señalados de Wilcoxon.

Las hipótesis estadísticas que se formularon para probar ambas técnicas son las siguientes:

$$H_0: \text{Med}_1 \geq \text{Med}_3, \quad \alpha = 0.05$$

$$H_a: \text{Med}_1 < \text{Med}_3$$

Donde Med_1 es la mediana del porcentaje de acierto obtenido con las técnicas ad hoc para el caso de estudio 1 y Med_3 , con UCPD para el caso de estudio 1. La tabla siguiente muestra los resultados obtenidos con esta prueba.

Tabla 6-17. Resultados de la prueba de los rango señalados de Wilcoxon de la técnicas ad hoc vs. UCPD (caso de estudio 1) para EC1-PROF

Variable	Resultado
V	63.500
Valor esperado	129.500
Varianza (V)	1247.875
P-valor (una cola)	0.032

Dado que el p-valor calculado es menor al nivel de significación $\alpha = 0.05$, se rechaza la hipótesis nula. Esto quiere decir que se puede afirmar, con un riesgo de equivocarse igual a 5%, que UCPD produjo mejores resultados que las técnicas ad hoc, en este caso de estudio. Esto corrobora las observaciones realizadas con el diagrama de cajas (ver Figura 6-10).

Como se puede observar en la Figura 6-11, los valores de las medianas son diferentes, aunque ambas están por encima del 80% de aciertos. Sin embargo, ambas cajas se encuentran en la misma posición y en similar rango de porcentajes de acierto (entre el 80% y 100% de aciertos). También, las diferencias entre los valores mínimo y máximo son muy similares. Por lo tanto, no se puede afirmar con qué caso de estudio en particular se produjeron mejores resultados al aplicar UCPD.

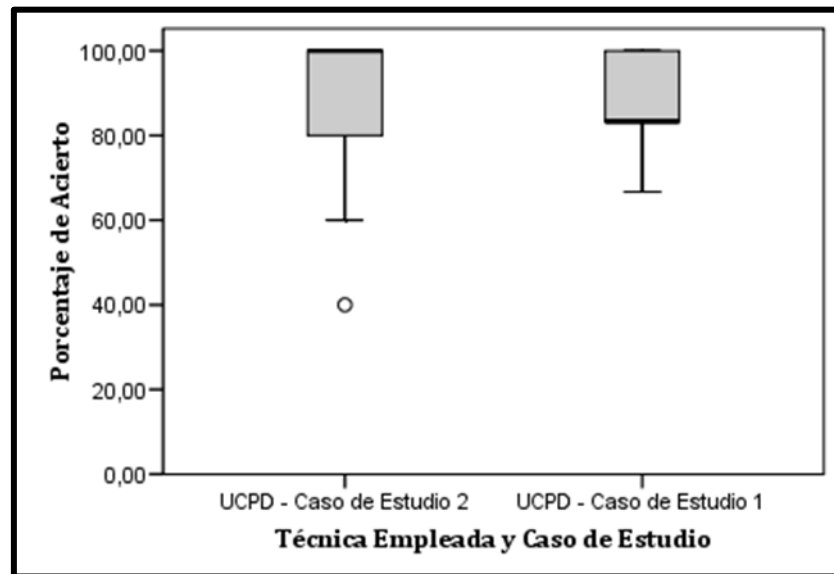


Figura 6-11. Diagrama de cajas con las diferencias entre la aplicación de UCPD con casos de estudio diferentes para EC1-PROF

Debido a los resultados observados en la Figura 6-11, se realizaron más pruebas estadísticas con la información recopilada en los cuestionarios. La Tabla 6-18 presenta la información estadística descriptiva al aplicar UCPD con los dos casos de estudio. La segunda columna de la tabla muestra los resultados obtenidos al aplicar UCPD con el caso de estudio correspondiente a un sistema de matrícula para un colegio de educación secundaria (caso de estudio 2) y la segunda columna de la tabla, los resultados obtenidos al aplicar UCPD con el caso de estudio correspondiente a un sistema para un restaurante (caso de estudio 1).

Tabla 6-18. Análisis estadístico descriptivo para los resultados de la aplicación de UCPD en diferentes casos de estudio para EC1-PROF

Variable	UCPD con Caso de Estudio 2	UCPD con Caso de Estudio 1
Observaciones	25	25
Mínimo	40.0%	66.67%
Máximo	100.00%	100.00%
Media	90.40%	87.33%
Desviación estándar	17.436	9.954

Como se puede observar en la tabla anterior (Tabla 6-18), las medias que se obtuvieron con los casos de estudio son muy parecidas. Para obtener resultados más concluyentes, se tuvieron que aplicar más pruebas estadísticas, las cuales se detallan a continuación.

Debido a que las muestras obtenidas al aplicar UCPD en los dos casos de estudio no siguen una distribución normal (ver la Tabla 6-12 para la prueba de normalidad para UCPD con el caso de estudio 2 y la Tabla 6-16 para la prueba de normalidad para UCPD con el caso de estudio 1), no se puede aplicar la prueba *t-student* de las muestras pareadas, por lo que se tiene que seleccionar una prueba no paramétrica. Debido al tipo de datos obtenidos, se emplea la prueba de los rangos señalados de Wilcoxon.

De manera similar a lo presentado en la sección previa y de acuerdo con el cuestionario 3, se definió una prueba para determinar si los resultados entre diferentes casos de estudio donde se empleó UCPD no son significativamente diferentes.

Las hipótesis estadísticas que se formularon para probar ambas técnicas son las siguientes:

$$H_0: Med_2 = Med_3, \quad \alpha = 0.05$$

$$H_a: Med_2 \neq Med_3$$

Donde Med_2 es la mediana del porcentaje de acierto obtenido con UCPD para el caso de estudio 1 y Med_3 , con UCPD para el caso de estudio 2. La Tabla 6-19 siguiente muestra los resultados obtenidos con esta prueba.

Tabla 6-19. Resultados de la prueba de los rango señalados de Wilcoxon para UCPD empleando los casos de estudio 1 y 2 para EC1-PROF

Variable	Resultado
V	191.000
Valor esperado	155.000
Varianza (V)	1331.625
p-valor (dos colas)	0.331

Dado que el p-valor calculado fue mayor al nivel de significación $\alpha=0.05$, no se pudo rechazar la hipótesis nula. Esto quiere decir que se puede afirmar, con un riesgo de equivocarse igual a 5%, que no hay diferencias significativas entre los resultados obtenidos con UCPD para diferentes casos de estudio.

b) Participantes que seleccionaron una secuencia diferente a maestros-transacciones-reportes

Fueron ocho los participantes de este estudio que seleccionaron una secuencia diferente a la de *maestros-transacciones-reporte* como la más fácil para construir software. Para los resultados obtenidos con estos participantes, no se pudo evaluar que UCPD produjo mejores resultados que las técnicas ad hoc, ya que la asunción de que la secuencia más fácil para construir software es *maestros-transacciones-reportes* no se cumple. Por lo tanto, no se pudo comparar los resultados obtenidos en el cuestionario 1, que corresponden a la aplicación de técnicas ad hoc, con los del cuestionario 2, que corresponden a la utilización de UCPD.

Sin embargo, sí se puede evaluar que UCPD es efectiva, asumiendo que la secuencia más fácil para construir software es *maestros-transacciones-reportes* (a esta secuencia se le denominará a partir de este momento como “secuencia asumida como verdadera”), aun si el desarrollador considera una secuencia diferente como la más sencilla para implementar un software.

Para realizar esta evaluación, solo se tomarán los datos obtenidos en el cuestionario 2, cuestionario en el que aplicaron UCPD para el caso de estudio 2. No se pudieron utilizar los datos obtenidos en el cuestionario 1, porque corresponden a las técnicas ad hoc y sus respuestas están influenciadas por la secuencia que consideran más fácil de construir de los participantes, la cual es diferente a la que se asumió para este estudio.

La Tabla 6-20 muestra los resultados obtenidos en el cuestionario 2 del estudio para estos participantes. El valor de “1” indica que el participante acertó en la respuesta y “0” que respondió de manera incorrecta. La primera columna muestra el número de participante, la segunda columna hasta la sexta presentan los resultados que obtuvo cada participante en cada pregunta y la última columna muestra el porcentaje de respuestas. Se tuvieron que descartar los

resultados de uno de los participantes, porque no respondió el cuestionario de acuerdo a lo obtenido con el UCPD que él mismo elaboró, a pesar de que lo realizó de manera correcta.

Tabla 6-20. Resultados obtenidos en el cuestionario 2 para los que seleccionaron otra secuencia para el estudio EC1-PROF

Participante	Pregunta					Porcentaje de Respuestas Correctas
	1	2	3	4	5	
1	1	1	1	1	1	100.0%
2	1	1	0	0	1	60.0%
3	0	1	1	0	1	60.0%
4	1	1	1	1	1	100.0%
5	0	1	1	1	1	80.0%
6	1	1	0	0	1	60.0%
7	1	1	1	1	0	80.0%

La Tabla 6-21 presenta la información estadística descriptiva obtenida luego de procesar los datos de los cuestionarios 2 de este estudio. La segunda columna de la tabla muestra los resultados obtenidos del cuestionario completado por los participantes que seleccionaron la secuencia asumida como verdadera para este estudio y la tercera columna, los que seleccionaron otra secuencia.

Tabla 6-21. Análisis estadístico descriptivo para el cuestionario 2 de todos los participantes del estudio EC1-PROF

Variable	Participantes que seleccionaron maestros-transacciones-reportes	Participantes que seleccionaron otra secuencia
Observaciones	25	7
Mínimo	40.0%	60.0%
Máximo	100.00%	100.00%
Media	90.40%	77.14%
Desviación estándar	17.436	17.995

Como se puede observar en la tabla anterior (Tabla 6-21), los valores máximos son iguales y los mínimos son muy cercanos, ya que uno es de 40% y el otro es de 60%. Aunque las medias son diferentes, su diferencia no es mucha, ya que difieren en solo 14%. Para determinar si los resultados obtenidos con

ambas muestras son similares, se tuvieron que aplicar más pruebas estadísticas, las cuales se detallan a continuación.

Debido a que la muestra obtenida con los participantes que seleccionaron que la secuencia asumida como verdadera no sigue una distribución normal (ver Tabla 6-12 para la prueba de normalidad de la muestra), se tuvo que aplicar una prueba no paramétrica. La prueba que se empleó fue la de Mann-Whitney.

Las hipótesis estadísticas que se formularon para probar ambas técnicas son las siguientes:

$$H_0: Med_2 = Med_4, \quad \alpha = 0.05$$

$$H_a: Med_2 \neq Med_4$$

Donde Med_2 es la mediana del porcentaje de acierto obtenido en el cuestionario 2 para los participantes que seleccionaron la secuencia de construcción asumida como verdadera y Med_4 , para los que seleccionaron otra secuencia. La tabla siguiente muestra los resultados obtenidos con esta prueba.

Tabla 6-22. Resultados de la prueba de Mann-Whitney para el cuestionario 2 para el estudio EC1-PROF

Variable	Resultado
U	124.500
Valor esperado	87.500
Varianza (V)	359.085
P-valor (una cola)	0.073

Dado que el p-valor calculado es mayor al nivel de significación $\alpha = 0.05$, no se pudo rechazar la hipótesis nula. Por lo tanto, se puede afirmar, con un riesgo de equivocarse igual a 5%, que UCPD produjo similares resultados en todos los participantes del estudio. Esto quiere decir que la técnica es efectiva, asumiendo que la secuencia más fácil para construir software es *maestros-transacciones-reportes*, aun si el desarrollador considera una secuencia diferente como la más sencilla para implementar un software.

6.3.2 EC2-POST: Segundo experimento controlado con estudiantes de posgrado

Debido a que en el experimento con profesionales EC1-PROF hubo participantes que no seleccionaron que la secuencia más fácil para construir casos de uso es *maestros-transacciones-reportes*, se replicó el experimento con

estudiantes de postgrado, quienes también eran profesionales con al menos dos años de experiencia en la industria.

En este experimento participaron 17 estudiantes de posgrado, 14 de los cuales eran profesionales con al menos 2 años de experiencia en proyectos de desarrollo de software. Muchos de ellos trabajaban en compañías que están mejorando sus procesos para construir software.

A diferencia de los participantes en la primera ejecución, no fue necesario realizar una clase de entrenamiento de casos de uso, ya que en habían tomado en el semestre anterior una asignatura en la que se revisaron todos los conceptos asociados a UML y, por consiguiente, ya sabían cómo especificar requisitos con casos de uso.

6.3.2.1 Materiales y casos de estudio

Los materiales empleados en el experimento fueron dos casos de estudio y tres cuestionarios. Los casos de estudio que se emplearon son los mismos a los que se utilizaron en el primer experimento controlado con profesionales: EC1-PROF. El primer caso de estudio corresponde a la elaboración de un sistema para un restaurante y el segundo, a un sistema de matrícula para un colegio de educación secundaria. Ambos casos de estudio son del tipo sistema de información. En el anexo A de este documento, se han incluido todos los documentos que se utilizaron en este experimento controlado

Los cuestionarios 1 y 2 corresponden a preguntas en lo que los participantes tenían que decidir entre dos casos de uso y responder cuál de ellos debería ser construido primero. En estos cuestionarios se incluyeron siete preguntas sobre secuencias de construcción, a diferencia de la cantidad de preguntas empleadas en la primera ejecución, la cual fue menor. Además, no se colocaron preguntas sobre secuencia de pruebas, porque el objetivo fue que los participantes se enfocaran sólo en la secuencia de construcción.

Al igual que en la primera ejecución, el objetivo del último cuestionario (cuestionario 4 de la primera ejecución y cuestionario 3 en esta ejecución) fue conocer la percepción de los participantes sobre UCPD. También, se incluyó una pregunta en la que tuvieron que seleccionar la forma más fácil para construir el software entre casos de uso de tipo maestro, transacción o reporte. En el anexo A se encuentran los cuestionarios y casos de estudio que se utilizaron en este experimento controlado.

6.3.2.2 Tareas realizadas durante el experimento

Se seleccionó el diseño de experimento intrasujeto, es decir, que los profesionales tuvieron que aplicar el primer caso de estudio con las técnicas ad hoc o técnicas propias y en el segundo, con UCPD. Se emplearon dos casos de estudio con similares características (ambos son sistemas de información), a fin de mitigar los efectos de aprendizaje. La Figura 6-12 muestra el diseño del experimento.

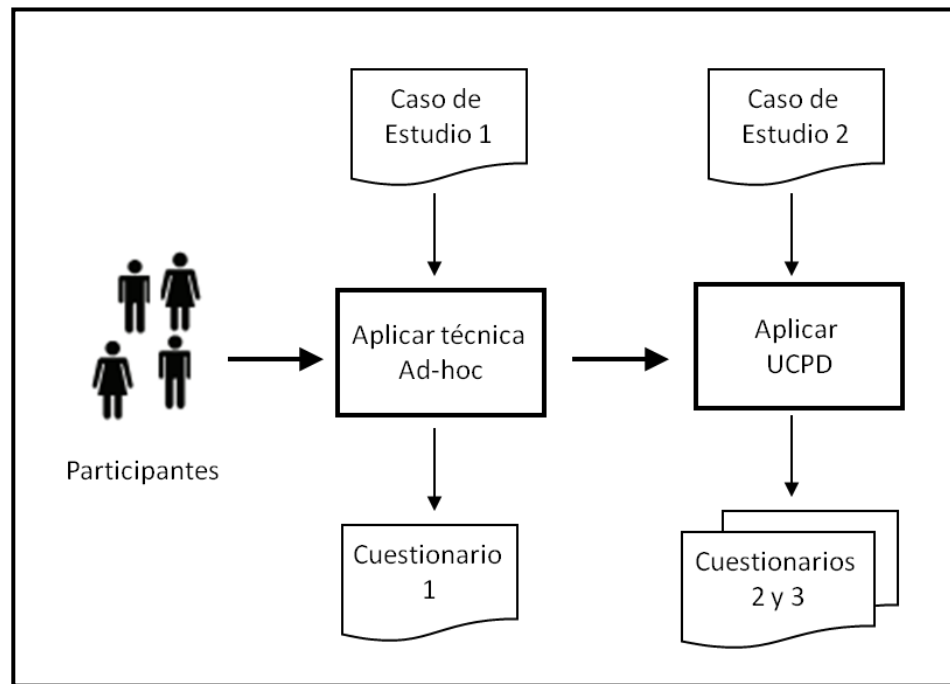


Figura 6-12. Diseño del segundo experimento controlado EC2-POST

A diferencia de la primera ejecución con profesionales, los participantes no aplicaron UCPD para el caso de estudio 1, ya que para fines estadísticos sólo era necesario que aplicaran la técnica en un caso de estudio. La sesión duró aproximadamente una hora y los profesionales emplearon 45 minutos en promedio para completar todas las tareas. Aunque no fue parte de este estudio conocer qué técnica demandó menor tiempo, se pudo observar que a los participantes no les tomó más de 10 minutos elaborar un UCPD. La Tabla 6-23 presenta las tareas realizadas durante la sesión por los participantes.

Tabla 6-23. Tareas realizadas por los participantes en EC2-POST

Tarea Nº	Descripción
1	Recibir casos de estudio 1 y cuestionario 1
2	Completar cuestionario 1
3	Recibir caso de estudio 2 y cuestionario 2
4	Elaborar UCPD para caso de estudio 2
5	Completar cuestionario 2
6	Completar cuestionario 3

6.3.2.3 Resultados cuantitativos

A diferencia de los resultados obtenidos en la primera ejecución, los participantes respondieron que la secuencia más fácil para construir casos de uso era maestros-transacciones-reportes.

La Tabla 6-24 y la Tabla 6-25 muestran los resultados obtenidos en los tres primeros cuestionarios del estudio. El valor de “1” indica que el participante acertó en la respuesta y “0” que respondió de manera incorrecta. La última columna muestra el total de respuestas que acertaron los participantes. La primera columna muestra el número de participante, la segunda columna hasta la séptima presenta los resultados que obtuvo cada participante en cada pregunta y la última columna muestra el total de respuestas correctas.

De los diecisiete participantes solo se consideraron los resultados de catorce, ya que tres estudiantes eran alumnos de pregrado que estaban tomando la asignatura en la que se realizó este experimento. Los alumnos de pregrado de la PUCP pueden tomar asignaturas de los programas de maestría, las cuales se considerarían dentro de sus optativas del plan de estudios.

Tabla 6-24. Resultados obtenidos en el cuestionario 1 para el estudio EC2-POST

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
1	1	0	0	1	1	0	0	3
2	1	0	0	1	1	1	0	4
3	1	0	0	1	0	0	1	3
4	0	1	0	1	1	0	1	4
5	1	0	0	1	1	0	0	3
6	1	0	1	1	1	1	1	6
7	1	0	1	1	0	0	1	4

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
8	1	1	1	1	1	0	1	6
9	1	0	1	1	1	1	0	5
10	1	1	0	0	1	0	0	3
11	0	1	0	1	0	0	1	3
12	1	1	1	1	1	1	1	7
13	1	0	0	0	1	0	1	3
14	1	0	0	0	1	0	0	2

Tabla 6-25. Resultados obtenidos en el cuestionario 2 para el estudio EC2-POST

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
1	1	1	1	1	1	1	1	7
2	1	1	1	1	1	1	1	7
3	1	1	1	0	1	1	1	6
4	1	0	1	1	1	0	1	5
5	1	1	1	1	1	1	1	7
6	1	0	1	1	1	1	1	6
7	1	1	1	1	1	1	1	7

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
8	1	1	1	1	1	1	1	7
9	1	1	1	1	1	1	1	7
10	1	1	1	1	0	0	1	5
11	1	1	0	0	0	0	0	2
12	1	0	1	1	0	1	1	5
13	1	1	0	1	1	1	1	6
14	0	1	0	1	1	1	1	5

La Figura 6-13 muestra el diagrama de cajas con los resultados obtenidos al aplicar las técnicas ad hoc (técnicas propias) de cada participante con el caso de estudio 1 y al aplicar UCPD con el caso de estudio 2.

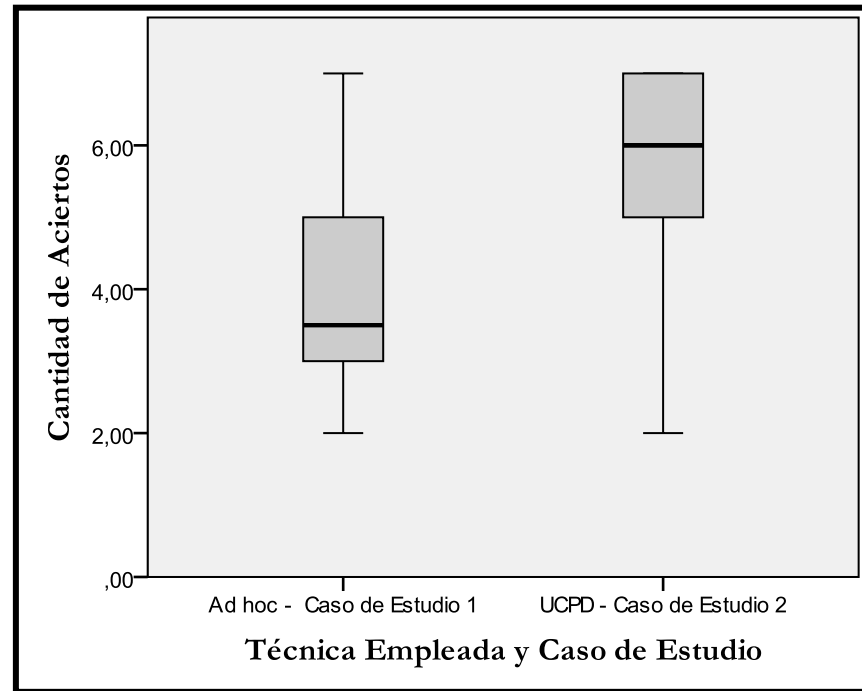


Figura 6-13. Diferencias entre la aplicación de técnicas ad hoc y de UCPD para EC2-POST

Como se puede observar en la Figura 6-13, el valor de las medianas es mayor al aplicar UCPD. Además, la caja correspondiente a UCPD se encuentra por encima de la caja de las técnicas ad hoc, aunque los valores mínimos y máximos (valores que se observan con los bigotes de las cajas) son iguales para ambas técnicas. También, el 75% de los datos al aplicar UCPD se encuentran entre 5, 6 ó 7, a diferencia con las técnicas ad hoc en las que sólo el 25% de los datos se encuentra en ese mismo rango. Según el diagrama, los resultados obtenidos al utilizar UCPD son mejores que al emplear las técnicas ad hoc.

A pesar de los resultados observados en la Figura 6-13, se realizaron más pruebas estadísticas con la información recopilada en los cuestionarios. La Tabla 6-26 presenta la información estadística descriptiva obtenida luego de procesar los datos de los cuestionarios que se muestran en la Tabla 6-24 y la Tabla 6-25. La segunda columna muestra los resultados obtenidos con las técnicas que los participantes emplean en su trabajo (técnicas ad hoc) y la tercera, los resultados con UCPD. Para las técnicas ad hoc se empleó el caso de estudio relacionado con un sistema de restaurantes (caso de estudio 1) y para UCPD se utilizó el caso de estudio relacionado con un sistema para un colegio de educación secundaria (caso de estudio 2).

Tabla 6-26. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para EC2-POST

Variable	Técnica Ad hoc con Caso de Estudio 1	UCPD con Caso de Estudio 2
Observaciones	14	14
Mínimo	2	2
Máximo	7	7
Media	4	5.857
Desv. Estándar	1.468	1.406

A fin de determinar si los resultados obtenidos con cada técnica siguen una distribución normal, se aplicó la prueba de Shapiro-Wilk. La Tabla 6-27 muestra los resultados obtenidos con este tipo de prueba.

Tabla 6-27. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para el estudio EC2-POST

Variable	Técnica Ad hoc con Caso de Estudio 1	UCPD con Caso de Estudio 2
W	0.871	0.774
P-valor	0.044	0.002

Dado que el p-valor calculado que se muestra en la tabla anterior para los resultados con UCPD fue menor que el nivel de significación establecido ($\alpha = 0.05$) se rechaza la hipótesis de distribución normal para las muestras obtenidas. Debido a estos resultados, se tuvo que aplicar la prueba no paramétrica de los rangos señalados de Wilcoxon.

Las hipótesis estadísticas formuladas para probar ambas técnicas son las siguientes:

$$H_0: \text{Med}_1 \geq \text{Med}_2, \quad \alpha = 0.05$$

$$H_a: \text{Med}_1 < \text{Med}_2$$

Donde Med_1 es la mediana del total de aciertos obtenido con las técnicas ad hoc y Med_2 , con UCPD.

Tabla 6-28. Resultados de la prueba de los rangos señalados de Wilcoxon de la técnica ad hoc vs. UCPD para el estudio EC2-POST

Variable	Resultado
V	9.0
Valor esperado	52.0
Varianza (V)	250.375
P-valor (una cola)	0.004

Como el p-valor computado es menor que el nivel de significación $\alpha=0.05$, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_a . Esto quiere decir que se pudo comprobar empíricamente que UCPD produjo mejores resultados (efectividad mayor) que los obtenidos con las técnicas ad hoc. Como se puede observar, estos resultados no difieren a los obtenidos en el estudio previo EC1-PROF.

6.3.3 EC3-PREG: Tercer experimento controlado con estudiantes de pregrado

En este experimento participaron 37 estudiantes de pregrado del cuarto año del programa de Ingeniería Informática del semestre 2010-1 de la PUCP. Este experimento fue una réplica del segundo experimento realizado con estudiantes de posgrado (profesionales).

Todos los estudiantes, a excepción de uno, respondieron que la secuencia más fácil para construir casos de uso era “maestros-transacciones-reportes”, por lo que este resultado no fue considerado. Además, un estudiante no pudo concluir el cuestionario del segundo caso de estudio. Finalmente, se pudo obtener información de sólo 35 estudiantes.

La Tabla 6-29 y la Tabla 6-30 muestran los resultados obtenidos en los dos primeros cuestionarios del estudio. El valor de “1” indica que el participante acertó en la respuesta y “0” que respondió de manera incorrecta. La primera columna muestra el número de participante, la segunda columna hasta la séptima presenta los resultados que obtuvo cada participante en cada pregunta y la última columna muestra el total de respuestas correctas por cada participante.

Tabla 6-29. Resultados obtenidos en el cuestionario 1 para EC3-PREG

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
1	1	1	0	1	1	0	0	4
2	1	0	1	0	0	1	0	3
3	1	1	1	1	1	1	1	7
4	0	1	1	1	0	1	1	5
5	1	1	1	1	1	1	1	7
6	0	1	1	1	0	1	0	4
7	1	0	1	0	1	0	1	4
8	1	0	0	1	1	0	0	3
9	1	1	1	0	1	0	0	4
10	1	0	0	1	0	0	0	2
11	0	1	1	1	0	0	1	4
12	1	1	1	1	1	1	1	7
13	1	0	1	1	0	1	1	5
14	0	1	1	1	1	1	1	6
15	1	1	0	1	1	0	1	5
16	1	1	1	1	1	0	0	5
17	0	1	1	1	1	0	0	4
18	0	1	0	1	1	0	1	4

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
19	1	1	1	1	1	0	1	6
20	1	1	1	1	1	0	0	5
21	1	1	1	1	1	0	0	5
22	1	0	0	1	0	0	0	2
23	1	0	0	0	1	0	0	2
24	1	0	1	1	1	1	1	6
25	1	1	1	1	0	1	0	5
26	1	1	0	1	1	1	0	5
27	1	1	1	1	0	1	1	6
28	1	0	0	1	1	0	0	3
29	1	1	1	1	0	1	0	5
30	1	1	1	1	0	1	1	6
31	0	1	1	1	0	1	1	5
32	1	0	1	1	0	1	1	5
33	1	1	1	1	1	1	1	7
34	1	1	0	1	1	1	0	5
35	0	0	1	1	1	1	1	5

Tabla 6-30. Resultados obtenidos en el cuestionario 2 para EC3-PREG

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
1	1	1	1	1	1	1	1	7
2	1	1	1	1	1	1	1	7
3	1	1	0	1	1	1	1	6
4	1	1	1	1	1	0	1	6
5	0	1	1	1	1	1	1	6
6	1	1	1	1	1	1	0	6
7	1	0	1	1	1	1	1	6
8	1	1	1	1	1	0	1	6
9	1	1	0	1	1	1	1	6
10	1	1	0	1	1	1	1	6
11	0	1	0	1	1	1	1	5
12	1	1	1	1	1	1	1	7
13	1	0	1	1	1	1	1	6
14	1	0	1	1	1	1	1	6
15	1	1	1	1	1	1	1	7
16	1	1	0	1	0	1	1	5
17	1	1	0	1	1	0	0	4
18	1	1	1	1	1	1	1	7

Participante	Pregunta							Total de Respuestas Correctas
	1	2	3	4	5	6	7	
19	1	1	1	1	1	1	1	7
20	1	1	1	1	1	0	0	5
21	1	1	1	1	1	1	1	7
22	1	1	0	0	1	1	0	4
23	1	1	0	1	1	0	1	5
24	1	1	1	1	1	0	1	6
25	1	1	1	1	1	1	1	7
26	1	1	0	1	1	1	1	6
27	1	1	1	1	1	1	1	7
28	1	1	1	1	1	1	1	7
29	1	1	1	1	1	1	1	7
30	1	0	0	0	1	1	1	4
31	1	1	1	1	1	1	1	7
32	1	1	0	1	0	0	0	3
33	1	1	1	1	1	1	1	7
34	1	1	1	1	1	1	1	7
35	1	1	1	1	1	0	1	6

La Figura 6-14 muestra el diagrama de cajas con los resultados obtenidos al aplicar las técnicas ad hoc (técnicas propias) de cada estudiante de pregrado con el caso de estudio 1 y al aplicar UCPD con el caso de estudio 2. Como se puede observar, el valor de las medianas es mayor al aplicar UCPD. Además, la caja correspondiente a UCPD se encuentra por encima de la caja de las técnicas ad hoc y la diferencia entre los valores mínimos y máximos (valores que se observan con los bigotes de las cajas) es menor también al aplicar UCPD (tomar en cuenta que el bigote superior en UCPD se superpone al valor máximo que es 7, la máxima cantidad de aciertos). También, el 75% de los datos al aplicar UCPD se encuentran entre 6 ó 7, a diferencia con las técnicas ad hoc en las que el 75% se encuentra desde 4 aciertos. Como se puede observar en el diagrama, los resultados obtenidos al utilizar UCPD son mejores que al emplear las técnicas ad hoc.

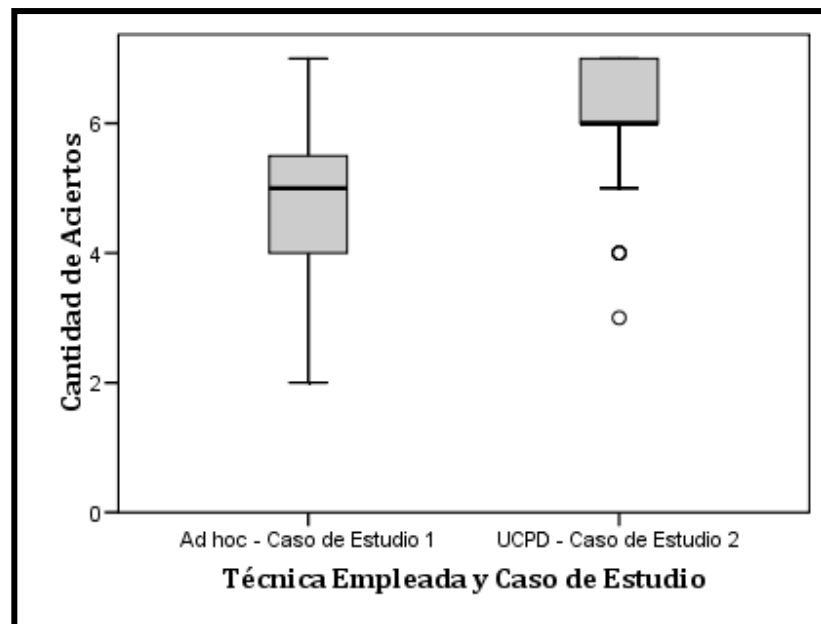


Figura 6-14. Diferencias entre la aplicación de técnicas ad hoc y de UCPD con diferentes casos de estudio para EC2-POST

Aunque los resultados observados en la Figura 6-14 muestran que se obtuvieron mejores resultados al aplicar UCPD, se realizaron más pruebas estadísticas con la información recopilada en los cuestionarios. La Tabla 6-31 presenta la información estadística descriptiva obtenida luego de procesar los datos de los cuestionarios que se muestran en la Tabla 6-29 y la Tabla 6-30. La segunda columna muestra los resultados obtenidos con las técnicas de los propios de los estudiantes de pregrado (técnicas ad hoc) y la tercera, los resultados con UCPD. Para las técnicas ad hoc se empleó el caso de estudio relacionado con un sistema de restaurantes (caso de estudio 1) y para UCPD se

utilizó el caso de estudio relacionado con un sistema para un colegio de educación secundaria (caso de estudio 2).

Tabla 6-31. Análisis estadístico descriptivo para la secuencia de construcción de casos de uso para EC3-PREG

Variable	Técnica Ad hoc con Caso de Estudio 1	Técnica UCPD con Caso de Estudio 2
Observaciones	35	35
Mínimo	2	3
Máximo	7	7
Media	4.743	6.029
Desv. Estándar	1.379	1.071

A fin de determinar si los resultados obtenidos con cada técnica siguen una distribución normal, se aplicó la prueba de Shapiro-Wilk. La Tabla 6-32 muestra los resultados obtenidos con este tipo de prueba.

Tabla 6-32. Resultados obtenidos con la prueba de normalidad de Shapiro-Wilk para EC3-PREG

Variable	Técnica Ad hoc con Caso de Estudio 1	UCPD con Caso de Estudio 2
W	0.925	0.02
P-valor	0.812	0.000

Debido a que el p-valor que se obtuvo (ver Tabla 6-33) es menor que el nivel de significación establecido ($\alpha = 0.05$), se rechaza la hipótesis de distribución normal para ambas muestras. Por ello, se tuvo que utilizar la prueba no paramétrica de los rangos señalados de Wilcoxon.

Las hipótesis estadísticas formuladas para probar ambas técnicas fueron:

$$H_0: \text{Med}_1 = \text{Med}_2, \quad \alpha = 0.05$$

$$H_a: \text{Med}_1 < \text{Med}_2$$

Donde Med_1 es la mediana del total de aciertos obtenido con las técnicas ad hoc y Med_2 , con UCPD.

Tabla 6-33. Resultados de la prueba de los rangos señalados de Wilcoxon de la técnica ad hoc vs. UCPD para EC3-PREG

Variable	Resultado
V	69.00
Valor esperado	301.00
Varianza (V)	3640.00
P-valor (una cola)	<0.0001

Dado que el p-valor computado es un menor que el nivel de significación $\alpha=0.05$, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_a . Esto quiere decir que se pudo comprobar empíricamente que UCPD produjo mejores resultados (efectividad mayor) que con las técnicas ad hoc para la muestra obtenida con los estudiantes de pregrado. Como se puede observar, estos resultados son similares a los que obtuvieron con los profesionales (estudio EC1-PROF) y con los estudiantes de posgrado (estudio EC2-POST).

6.3.4 Amenazas a la validez

En esta sección se discuten las diferentes amenazas en la validez de este estudio y cómo se intentaron aliviar.

6.3.4.1 Validez de la construcción

La validez de la construcción es el grado con el cual las variables independientes y dependientes son medidas con precisión por los instrumentos empleados en el estudio.

Los cuestionarios permitieron medir cuantitativamente la efectividad de las técnicas a comparar a través de la cantidad de aciertos de las preguntas respondidas por los participantes. Además, las preguntas consideraron la selección de la construcción entre los siguientes pares de tipos de casos de uso: maestros-transacciones, transacciones-transacciones y maestros-reportes. Esto permitió comparar los resultados obtenidos para ambos casos de estudio.

6.3.4.2 Validez interna

La validez interna es el grado con el cual las conclusiones pueden ser extraídas de la relación causa-efecto de las variables independientes sobre las variables dependientes. Además, en función de los resultados de la experimentación, se puede concluir empíricamente que existe una relación entre las variables independientes y las dependientes. Los diferentes aspectos que pudieron amenazar la validez interna son los siguientes:

- **Diferencia entre participantes.** En el caso del primer experimento controlado (EC1-PROF), los participantes tenían diferente grado de conocimiento y experiencia. Además, algunos de ellos no empleaban casos de uso en sus trabajos, aunque sí sabían de gestión de requisitos. Para minimizar este problema se realizó una sesión de entrenamiento de casos de uso cuatro días antes de la ejecución del experimento. En el segundo experimento controlado (EC2-POST) y en el tercero (EC3-PREG), los participantes tenían el mismo grado de conocimiento y experiencia debido a que pertenecían a una misma asignatura y ya habían aprendido UML en asignaturas previas.
- **Efectos de aprendizaje.** La aplicación de casos de estudio diferentes canceló el efecto de aprendizaje, efecto que podría haberse producido si sólo se hubiera aplicado un caso de estudio.
- **Conocimiento.** Se emplearon los mismos casos de estudio para todos los sujetos, los cuales eran del tipo sistema de información.
- **Efectos de fatiga.** A cada participante le tomó aproximadamente una hora en promedio para aplicar ambos casos de estudio y responder los cuestionarios. Por lo tanto, la fatiga no fue relevante.
- **Efectos de persistencia.** Los participantes no habían hecho un experimento similar anteriormente.
- **Motivación de los sujetos.** Los profesionales y estudiantes de posgrado estuvieron motivados porque querían conocer sobre nuevas técnicas para mejorar su trabajo. Además, alguno de los profesionales que participaron estaban trabajando en compañías que se estaban preparando para obtener el nivel 3 de CMMI o ya lo habían obtenido. Los estudiantes de pregrado también estuvieron motivados, porque tenían que emplear UCPD en el proyecto de la asignatura de Ingeniería de Software.

6.3.4.3 Validez externa

La validez externa es el grado con el que los resultados se pueden generalizar a otros contextos. Una de las amenazas que podría limitar la habilidad de realizar una generalización sobre el estudio es el material empleado. En el experimento se trató de utilizar casos de estudio que pudieran ser representativos de casos reales. Aunque los participantes provenían de entornos empresariales, se necesitarían más estudios empíricos con casos reales provenientes de la industria.

Otro aspecto que podría limitar la generalización es el hecho de que algunos profesionales consideraron que la secuencia más sencilla para realizar la secuencia de construcción, desde la perspectiva del desarrollador, fue diferente a la secuencia *maestros-transacciones-reportes*. Desde el punto de vista de algunos profesionales, el mantenimiento de maestros no es relevante, pero sí sería útil si sólo se incluyera en el diagrama los casos de uso correspondientes a transacciones y reportes.

6.3.5 Discusión de los resultados

Los resultados obtenidos en los experimentos controlados con alumnos de pregrado y profesionales muestran que se obtienen mejores resultados al aplicar UCPD para determinar la secuencia de construcción de casos de uso que al emplear técnicas propias o ad hoc.

El primer experimento controlado fue realizado con profesionales que tienen al menos dos años de experiencia en proyectos de desarrollo de software, muchos de ellos trabajan en empresas que han obtenido la certificación de nivel 3 de CMMI o estuvieron en proceso de evaluación en el momento que se realizó el experimento. En este estudio, la mayoría de participantes seleccionaron que la secuencia que es más fácil para construir un software era *maestros-transacciones-reportes*, y, con los datos obtenidos con estos participantes, se pudo demostrar que UCPD produce mejores resultados que al emplear las técnicas propias o ad hoc. También se procesaron los datos obtenidos con aquellos profesionales que seleccionaron otra secuencia de construcción y, para ellos, se pudo comprobar que UCPD sigue siendo efectiva, siempre y cuando se asuma que la secuencia más fácil para construir software es *maestros-transacciones-reportes*.

Aunque los resultados obtenidos en el primer experimento controlado fueron satisfactorios, además a las preguntas sobre secuencia de construcción de casos de uso (diseño y programación) se incluyeron preguntas sobre la secuencia de realización de pruebas. Estas preguntas adicionales podrían haber influido en el resultado del experimento controlado, por ello se realizó una réplica de este experimento con las siguientes variantes: sólo preguntas de secuencia de construcción, la misma cantidad de preguntas para cada caso de estudio (en el primer experimento habían seis preguntas para el caso de estudio 1 y cinco para el caso de estudio 2) y no aplicaron UCPD en el caso de estudio 1, que era el que habían empleado para aplicar la técnica propia o técnica ad hoc.

El segundo experimento controlado fue aplicado a estudiantes de posgrado quienes también tenían experiencia de al menos dos años en la industria, por lo que se podría considerar que era una muestra similar al del primer experimento controlado. Los resultados obtenidos fueron satisfactorios, por lo que se comprueba empíricamente que UCPD provee mejores resultado que si se aplicaran las técnicas ad hoc.

Además, se realizó un tercer experimento controlado con estudiantes de pregrado. El material empleado fue el mismo que el utilizado en el segundo experimento controlado con alumnos de posgrado. Los resultados obtenidos fueron satisfactorios, de manera similar a los obtenidos con los profesionales y los estudiantes de posgrado. Esto podría indicarnos que no hubiera sido necesario realizar estudios con profesionales para comprobar empíricamente que UCPD provee mejores resultados con las técnicas ad hoc, aunque una amenaza muy fuerte para la validez de estos resultados sería que los alumnos de pregrado tienen muy poca experiencia en proyectos de software y no conocen otra técnica equivalente porque aún no la han necesitado.

6.4 Evaluación de la percepción de UCPD de los experimentos controlados

Con el fin de evaluar la percepción de participantes de los experimentos controlados sobre UCPD (profesionales y estudiantes de pregrado), se incluyeron en el cuestionario preguntas similares a las realizadas en la evaluación inicial con alumnos de pregrado, en las que tuvieron que opinar sobre tres aspectos: facilidad de uso, utilidad e intención de uso de la técnica en otros proyectos.

Empleando el criterio "PICOC" propuesto por Petticrew y Roberts [Petticrew, 2006], esta evaluación se definió de la siguiente manera:

Población (*population*): desarrollo de software basado en casos de uso

Intervención (*intervention*): la técnica UCPD.

Comparación (*comparison*): técnicas ad hoc o propias

Salidas (*outcomes*): percepción positiva de UCPD, según las variables definidas en el MAM.

Contexto (*context*): profesionales con al menos 2 años de experiencia en proyectos de desarrollo de software y estudiantes del cuarto año de Ingeniería Informática de la Pontificia Universidad Católica del Perú quienes han podido comparar sus técnicas propias o ad hoc con UCPD.

Las preguntas de investigación que se han identificado son las mismas que para la evaluación inicial con los alumnos de pregrado (ver sección 6.2):

- ¿UCPD es percibida como fácil de usar?
- ¿UCPD es percibida como útil?
- ¿Existe la intención de usar UCPD en futuros proyectos?

6.4.1 Evaluación con profesionales

El cuestionario que se diseñó tuvo algunas variaciones de forma con respecto al empleado con los alumnos de pregrado en el estudio EI-PREG con el objetivo que éste sea más fácil y rápido de responder. El cuestionario fue aplicado al final del experimento controlado con profesionales (EC1-PROF) y del experimento controlado con estudiantes de posgrado (EC2-POST), cuyos resultados cuantitativos se muestran en la sección 6.3.

La Figura 6-15 muestra una vista parcial del cuestionario empleado para la evaluación inicial con los alumnos del pregrado y la Figura 6-16 el que se utilizó para los profesionales (los cuestionarios completos y la información que registraron los participantes se encuentran en el Anexo A). Cabe resaltar que el

diseño de este cuestionario es una adaptación del estudio presentado en la tesis doctoral de Abrahao [Abrahao, 2004].

5. Marque con una "X" el círculo que esté más cercano a su opinión.

a. Me parece difícil y complejo construir el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Me parece fácil y sencillo construir el diagrama de precedencias
b. Creo que el diagrama de precedencias NO es útil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Creo que el diagrama de precedencias es muy útil para planificar las iteraciones de construcción

Figura 6-15. Vista parcial del cuestionario empleado para los alumnos de pregrado en EI-PREG

DIAGRAMA DE PRECEDENCIAS	-				+
3. Facilidad para construir o realizar el diagrama de precedencias	0	1	2	3	4
4. Utilidad del diagrama de precedencias para determinar la secuencia de construcción de software	0	1	2	3	4

Figura 6-16. Vista parcial del cuestionario empleado para los profesionales en EC1-PROF

Como se puede observar en las figuras anteriores, aunque en ambos cuestionarios se indican preguntas relacionadas a facilidad de uso, utilidad e intención de uso, el diseño del segundo cuestionario contiene menos texto. Esto se hizo con la intención que fuera algo menos complicado para responder. Además, ambos cuestionarios utilizan una escala de cinco niveles. La Tabla 6-34 muestra las preguntas que se incluyeron en el cuestionario y al tipo de variable que corresponde.

Tabla 6-34. Preguntas por cada constructor del MAM empleadas en EC1-PROF y EC2-POST

Constructor del MAM	Pregunta
Facilidad de uso	2. Facilidad para construir o realizar el diagrama de precedencias
Utilidad	3. Utilidad del diagrama de precedencias para determinar la secuencia de construcción de software.
Intención de uso	6. Usaría el diagrama de precedencias para siguientes proyectos de desarrollo de software

Al igual que para los alumnos de pregrado, la hipótesis estadística para evaluar cada pregunta son las siguientes:

$$H_0: \mu \leq 3, \alpha = 0.05$$

$$H_a: \mu > 3$$

Donde α representa el 5% de significación, esto quiere decir que existe un 0.05 de probabilidad de aceptar la hipótesis alternativa cuando la hipótesis nula es verdadera (error de tipo I). Para el análisis estadístico se empleó la herramienta SPSS. A continuación se muestran los resultados obtenidos para las preguntas sobre facilidad de uso, utilidad e intención de uso.

a. Facilidad de uso percibida

En cuanto a la pregunta de investigación sobre facilidad de uso de UCPD, la Tabla 6-35 presenta el resultado del análisis descriptivo realizado para los datos obtenidos de la pregunta sobre facilidad de uso.

Tabla 6-35. Resultados sobre facilidad de uso percibida con profesionales, muestras de EC1-PROF y EC2-POST

Variable	Resultados para EC1-PROF	Resultados para EC2-POST
Observaciones	25	14
Mínimo	2	3
Máximo	5	5
Media	3.84	4
Desviación Estándar	0.850	0.679

Se puede observar que las medias son mayores a 3.5, la cual es superior al puntaje medio (3). El rango de valores osciló entre 2 y 5 para EC1-PROF, y entre 3 y 5 para EC2-POST. Como los datos obtenidos son de tipo ordinal se tuvo que emplear la prueba no paramétrica de los rangos señalados de Wilcoxon para una muestra, pero como esta prueba emplea la mediana y no la media, se considerarán las hipótesis estadísticas definidas cambiando la media (μ) por mediana. Con esta prueba, se obtuvo un estadístico W igual a 194 para EC1-PROF y 66 para EC2-POST. Puesto que el p-valor para la prueba que se obtuvo para ambas muestras fue menor que 0.05 (p-valor menor a 0.003), pudo rechazarse la hipótesis nula. Por lo tanto, podemos afirmar, con un riesgo de equivocarse igual a 5%, que UCPD fue percibida como fácil de usar. Esto quiere decir que los profesionales percibieron que UCPD es fácil de usar.

b. Utilidad percibida

La Tabla 6-36 presenta el resultado del análisis descriptivo realizado para los datos obtenidos de las preguntas sobre utilidad percibida. Se puede observar que la media para ambas muestras es mayor a 4.2, la cual es superior al puntaje medio (3). El rango de valores osciló entre 3 y 5.

Tabla 6-36. Resultados sobre utilidad percibida con profesionales

Variable	Resultados para EC1-PROF	Resultados para EC2-POST
Observaciones	25	14
Mínimo	3	3
Máximo	5	5
Media	4.36	4.214
Desviación Estándar	0.638	0.699

Debido a que los datos obtenidos de la muestra son de tipo ordinal, se utilizó la prueba de los rangos señalados de Wilcoxon para una muestra y se obtuvo un estadístico W igual a 276 para EC1-PROF y 78 para EC2-POST. Como el p-valor obtenido para ambas muestras fue menor que 0.05 (p-valor < 0.003 para ambas), pudo rechazarse la hipótesis nula (para las hipótesis estadísticas se empleó la mediana y no la media). Por lo tanto, se pudo demostrar empíricamente, con un riesgo de equivocarse igual a 5%, que los profesionales piensan que UCPD es útil.

c. Intención de uso

La Tabla 6-37 muestra el resultado del análisis descriptivo realizado para los datos obtenidos de las preguntas sobre intención de uso. Se puede observar que las medias son mayores a 4, las cuales son superiores al puntaje medio (3). El rango de valores para EC1-PROF osciló entre 1 y 5; sin embargo el valor mínimo para EC2-POST fue 3, el cual es el valor medio.

Tabla 6-37. Resultados sobre intención de uso con profesionales

Variable	Resultados para EC1-PROF	Resultados para EC2-POST
Observaciones	25	14
Mínimo	1	3
Máximo	5	5
Media	4.2	4.214
Desviación Estándar	0.913	0.579

De manera similar al análisis realizado para la facilidad de uso y utilidad percibidas, se tuvo que aplicar la prueba de los rangos señalados de Wilcoxon. Con esta prueba se obtuvo un estadístico W igual a 258 para EC1-PROF y 91 para EC2-POST. Puesto que el p -valor para las pruebas de ambas muestras fue menor que 0.05 (el p -valor fueron menores a 0.003), pudo rechazarse la hipótesis nula con un 95.0% de nivel de confianza (para las hipótesis estadísticas se empleó la mediana y no la media). Por lo tanto, podemos afirmar, con un riesgo de equivocarse igual a 5%, que existe una intención, por parte de los profesionales de usar UCPD.

6.4.2 Evaluación con estudiantes de pregrado

El cuestionario que se diseñó tuvo algunas variaciones al empleado con los profesionales, ya que se incluyeron conceptos sobre lo que significaba utilidad dentro el cuestionario. La Figura 6-17 muestra una vista parcial del cuestionario empleado para los alumnos del pregrado (los cuestionarios completos y la información que registraron los participantes se encuentran en el Anexo A).

DIAGRAMA DE PRECEDENCIAS					
	-				+
2. Facilidad para construir o realizar el diagrama de precedencias	0	1	2	3	4
3. Utilidad del diagrama de precedencias para determinar la secuencia de construcción de software (Utilidad es el grado en el cual UCPD será efectivo para lograr los objetivos propuestos; es decir determinar la secuencia de construcción de software)	0	1	2	3	4
4. ¿Usaría el diagrama de precedencias para siguientes proyectos de desarrollo de software?	0	1	2	3	4

Figura 6-17. Vista parcial del cuestionario empleado en el experimento controlado en alumnos de pregrado EC3-PREG

La hipótesis estadística empleada fue la misma que la utilizada con los profesionales (ver subsección 6.4.1)

a. Facilidad de uso percibida

En cuanto a la pregunta de investigación sobre facilidad de uso de UCPD, la Tabla 6-38 presenta el resultado del análisis descriptivo realizado para los datos obtenidos de la pregunta sobre facilidad de uso.

Tabla 6-38. Resultados sobre facilidad de uso percibida en EC3-PREG

Variable	Resultado
Observaciones	35
Mínimo	3
Máximo	5
Media	3.971
Desviación Estándar	0.618

Se puede observar que la media es de 3.971, la cual es superior al puntaje medio (3). El rango de valores osciló entre 3 y 5, lo que quiere decir que ningún alumno evaluó la técnica con un puntaje menor a 3, que es el puntaje medio. Debido a que los datos obtenidos son de tipo ordinal, se tuvo que aplicar la prueba no paramétrica de los rangos señalados de Wilcoxon.

Al emplear la prueba de los rangos señalados de Wilcoxon para una muestra, se obtuvo un estadístico W igual a 406. Puesto que el p -valor para la prueba es menor que 0.05 (p -valor < 0.001), pudo rechazarse la hipótesis nula, con un riesgo de equivocarse igual a 5% (para las hipótesis estadísticas se empleó la mediana y no la media). Esto quiere decir que los estudiantes de pregrado pensaron que UCPD fue fácil de usar.

b. Utilidad percibida

La Tabla 6-39 presenta el resultado del análisis descriptivo realizado para los datos obtenidos de las preguntas sobre utilidad percibida. Se puede observar que la media es de 4.486, la cual es muy superior al puntaje medio (3). El rango de valores osciló entre 3 y 5, lo que quiere decir que ningún alumno evaluó la técnica con un puntaje menor al puntaje medio.

Tabla 6-39. Resultados sobre utilidad percibida en EC3-PREG

Variable	Resultado
Observaciones	31
Mínimo	3
Máximo	5
Media	4.486
Desviación Estándar	0.658

El estadístico W, según la prueba de los rangos señalados de Wilcoxon, fue igual a 528 (no se pudo aplicar pruebas paramétrica para este caso porque los datos obtenidos son de tipo ordinal y para las hipótesis estadísticas se empleó la mediana y no la media). Puesto que el p-valor para la prueba es menor que 0.05 (p-valor<0.001), pudo rechazarse la hipótesis. Por lo tanto, podemos, con un riesgo de equivocarse igual a 5%, que UCPD fue percibida como útil.

c. Intención de uso

La Tabla 6-40 muestra el resultado del análisis descriptivo realizado para los datos obtenidos de las preguntas sobre intención de uso. Se puede observar que la media es de 4.4, la cual es muy superior al puntaje medio (3). El rango de valores osciló entre 2 y 5.

Tabla 6-40. Resultados sobre intención de uso en EC3-PREG

Variable	Resultado
Observaciones	35
Mínimo	2
Máximo	5
Media	4.4
Desviación Estándar	0.736

Al igual que las muestras obtenidas con los alumnos de pregrado en relación a la facilidad de uso y a la utilidad percibida, se tuvo que utilizar la prueba no paramétrica de los rangos señalados de Wilcoxon, porque los datos obtenidos son de tipo ordinal.

Al emplear la prueba de los rangos señalados de Wilcoxon para una muestra, se obtuvo un estadístico W igual a 553. Puesto que el p-valor para la prueba fue menor que 0.05 (p-valor <0.001), pudo rechazarse la hipótesis nula con un 95.0% de nivel de confianza (para las hipótesis estadísticas se empleó la mediana y no la media). Por lo tanto, podemos afirmar, con un riesgo de equivocarse igual a 5%, que existe la intención de usar UCPD por parte de los alumnos de pregrado.

6.4.3 Discusión de los resultados

Los resultados obtenidos muestran de manera general una actitud positiva de los profesionales y de los estudiantes hacia la técnica. Los participantes percibieron que UCPD es fácil de usar y útil, y que también la emplearían en futuros proyectos de software.

Se cambió el diseño del cuestionario que fue utilizado por los alumnos de pregrado, en el estudio EC3-PREG para que sea más fácil de completar. Al igual que con el cuestionario empleado en la evaluación inicial con estudiantes de pregrado (estudio EI-PREG), se podría considerar como inconveniente el haber utilizado una pregunta por constructor, pero, tal y como se comentó en la sección 6.2.4, existen estudios que han empleado un cuestionario con características similares al empleado sin que se invaliden los resultados obtenidos.

Se quiso hacer un cuestionario fácil de completar, ya que los dos experimentos controlados con profesionales (EC1-PROF y EC2-POST) se realizaron después de las 7.30pm, luego de finalizar la jornada laboral de los participantes, por lo que no se quería alargar mucho la sesión y de esta manera reducir al máximo el tedio y el cansancio que podría producir en ellos el experimento.

Como se puede observar, los resultados obtenidos sobre la percepción de UCPD son muy similares a los obtenidos en el estudio EI-PREG correspondiente a la evaluación inicial con alumnos de pregrado (ver sección 6.2).

6.5 Evaluación del MAM

El objetivo de esta sección es evaluar la relación entre las variables del MAM propuestas por Moody y que se muestran en la Figura 6-1.

Utilizando la plantilla *Goal/Question/Metric* (GQM) propuesta por Basili [Basili, 1994], esta evaluación se definió de la siguiente manera

Analizar: las relaciones entre las variables propuestas por el MAM

Con el propósito de: evaluar

Con relación a: su nivel de influencia/dependencia/causalidad

Desde el punto de vista del: investigador

En el contexto de: profesionales con al menos 2 años de experiencia y de estudiantes de pregrado del cuarto año de Ingeniería Informática de la Pontificia Universidad Católica del Perú quienes han empleado UCPD como parte de un experimento controlado o en un proyecto de desarrollo de software y han respondido un cuestionario según las variables propuestas por el MAM.

Según la propuesta de Moody, existe una relación entre las siguientes variables:

- La percepción de utilidad es influenciada por la facilidad de uso percibida
- La intención de uso es influenciada por la facilidad de uso percibida
- La intención de uso es influenciada por la utilidad percibida.

Para evaluar la relación entre las variables, se empleará en primer lugar el coeficiente de correlación, de manera similar a los estudios realizados por Davis [Davis, 1989] y Adams et al. [Adams, 1993].

Según Muijs [Muijs, 2004] para determinar si existe un grado de relación entre dos variables de tipo ordinal no se debe utilizar el coeficiente de correlación de Pearson, sino el de correlación de Spearman. La Tabla 6-41 muestra las reglas generales para indicar la fortaleza entre las relaciones de dos variables propuesta por este autor.

Tabla 6-41. Reglas para determinar la fortaleza de la relación entre dos variables

Rango del valor absoluto del índice de correlación	Fortaleza de la relación
[0.0 , 0.1]	Débil
[0.1 , 0.3]	Modesto
[0.3 , 0.5]	Moderado
[0.5 , 0.8]	Fuerte
[0.8 , 1.0]	Muy fuerte

Adicionalmente, para aquellas variables que tengan un coeficiente de correlación mayor o igual a moderado (calificación según la tabla anterior), se ha empleado un análisis de regresión, de manera similar a un estudio incluido en la tesis doctoral de Abrahao [Abrahao, 2004]. Como los cuestionarios emplean una escala de Likert y generalmente hay una pregunta por cada variable, éstas no son continuas; por lo tanto no se puede realizar una regresión lineal, sino ordinal según la propuesta de McCullagh y Nelder [McCullagh, 1989]. Aunque hay autores, como Norris et. al [Norris, 2006], que sugieren que se pueden obtener buenos resultados con modelos de regresión lineal para datos de tipo ordinal, se ha preferido seguir las sugerencias McCullagh y Nelder.

La función de enlace es una transformación de las probabilidades acumuladas que permiten la estimación de un modelo de regresión ordinal. Para poder determinar un modelo de este tipo se tiene que escoger una de estas

funciones de enlace, tomando en cuenta la distribución de valores de la variable dependiente. La Tabla 6-42 muestra qué función de enlace se sugiere emplear según la distribución de valores que sigue la variable dependiente.

Tabla 6-42. Funciones de enlace por distribución de variable dependiente

Función de Enlace	Distribución de Valores de la Variable Dependiente
Logit	Categorías distribuidas de manera uniforme
Log-log complementario	Categorías más altas, más probables
Log-log negativo	Categorías más bajas, más probables
Probit	La variable sigue una distribución normal
Cauchit (Cauchy inversa)	La variable tiene muchos valores extremos

También, otro de los objetivos de esta evaluación es comparar resultados entre los alumnos de pregrado y los profesionales.

6.5.1 Evaluación con estudiantes de pregrado

La Figura 6-18 muestra el coeficiente de correlación de Spearman, también conocido como rho de Spearman (ρ), entre las variables del MAM para las muestras obtenidas con los alumnos de pregrado. Además, se indica la fortaleza de las relaciones según la Tabla 6-41.

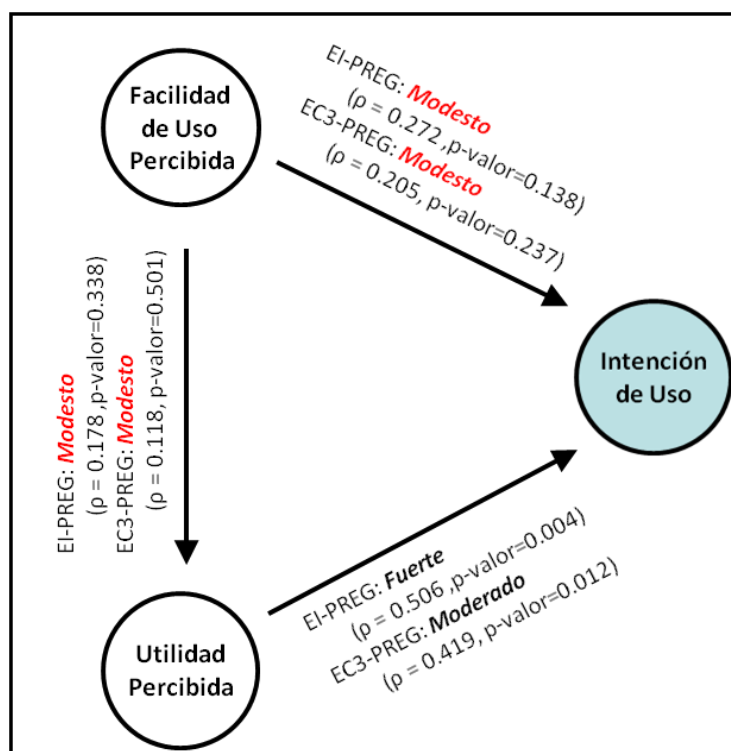


Figura 6-18. Fortaleza de las relaciones del MAM para estudiantes de pregrado

Como se puede observar, la correlación entre la facilidad de uso percibida e intención de uso es modesta para ambas muestras obtenidas con alumnos de pregrado (evaluación inicial EI-PREG y experimento controlado EC3-PREG). Esto quiere decir, que no podemos demostrar empíricamente que haya una dependencia entre ambas variables o que una de ellas pueda influir sobre la otra con las muestras obtenidas.

En cuanto a la correlación entre facilidad de uso percibida y la utilidad percibida, como se puede observar, la correlación entre ambas variables es modesta para las dos muestras. Esto significa que no podemos demostrar empíricamente que hay una dependencia entre ambas variables o que una de ellas pueda influir sobre la otra con las muestras obtenidas.

Sin embargo, la correlación entre utilidad percibida y la intención de uso es fuerte para la muestra obtenida en la evaluación inicial EI-PREG y moderada para la muestra del experimento controlado EC3-PREG. Por lo tanto, existe una dependencia entre ambas variables. Para confirmar la relación causal entre estas variables, se empleó el análisis de regresión ordinal, el cual se incluye a continuación.

6.5.1.1 Modelo de regresión para la relación utilidad percibida-intención de uso para EI-PREG

Para determinar el modelo de regresión ordinal para la muestra obtenida en la evaluación inicial con estudiantes de pregrado EI-PREG, se empleó como variable independiente a la intención de uso (IU) y como variable dependiente a la utilidad percibida (UP). La Figura 6-19 muestra la distribución de valores para la variable dependiente IU.

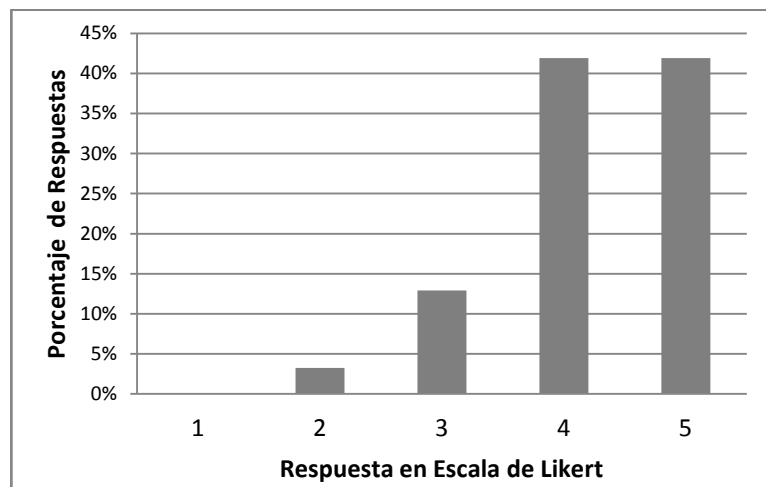


Figura 6-19. Distribución de valores de IU en EI-PREG

Según la información de la Tabla 6-42, la función de enlace más adecuada sería la Log-log complementaria, porque hay muchos valores en las categorías más altas. Sin embargo, se obtuvieron mejores resultados con la función Cauchit.

Tabla 6-43. Información sobre el ajuste de modelos para UP vs. IU en EI-PREG

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Solo intersección	30.252			
Final	17.581	12.671	2	0.02

Según la Tabla 6-43, el estadístico chi-cuadrado de la regresión es de 12.116 con 2 grados de libertad, siendo el p-valor de 0.002. Estos valores indican que el modelo obtenido con la función de enlace Cauchit provee una mejora significativa que un modelo de sólo intersección. Esto quiere decir que el modelo de regresión ordinal obtenido es altamente significativo y provee

mejores predicciones que sólo adivinar basándose en las probabilidades marginales de las categorías de respuesta sobre utilidad. Además, el modelo obtenido confirma que la utilidad percibida tiene un efecto significativo en la intención de usar UCPD en los alumnos de pregrado que participaron en la evaluación inicial. La Tabla 6-44 muestra el estadístico chi-cuadrado de Pearson y el estadístico chi-cuadrado basado en la desviación. El propósito de estos estadísticos es probar si los datos observados son inconsistentes con el modelo ajustado.

Tabla 6-44. Bondad de ajuste para UP vs. IU en EI-PREG

	Chi-cuadrado	gl	p-valor
Pearson	6.224	4	0.183
Desviación	6.447	4	0.168

Como se puede observar en la tabla anterior, los p-valor obtenidos son significativos. Esto quiere decir que los datos y las predicciones del modelo son similares, por lo que se ha obtenido un buen modelo. La Tabla 6-45 muestra la información del modelo de regresión ordinal empleando la función Log-log complementaria.

Tabla 6-45. Estimación de los parámetros UP vs IU en EI-PREG

		Estimación	Error típ.	gl	p-valor
Umbral	[IU = 2]	-10.243	6.544	1	0.118
	[IU = 3]	-4.052	1.918	1	0.035
	[IU = 4]	-0.179	0.383	1	0.640
Ubicación	[UP = 3]	-6.113	3.429	1	0.075
	[UP = 4]	-1.665	1.279	1	0.193
	[UP = 5]	0	.	0	.

Según la tabla anterior y lo indicado por Borooah [Borooah, 2001], para obtener la intención de uso (IU) basada en la utilidad percibida (UP) y la facilidad de uso percibida, se necesita calcular lo siguiente: $D = -6.113 \times UP_3 - 1.665 \times UP_4$; donde UP_3 es uno, si la utilidad percibida toma el valor de 3 y 0

en caso contrario; y UP_4 es uno, si la utilidad percibida es 4 y 0 en caso contrario. La IU se determina con el **D** calculado:

- IU = 2, si $D < -10.243$
- IU = 3, si $-10.243 \leq D < -4.052$
- IU = 4, si $-4.052 \leq D < -0.179$
- IU = 5, si $D \geq -0.179$

Para determinar si se cumple la suposición de que los parámetros de ubicación (coeficientes para las pendientes) son los mismos para todas las categorías de respuesta para el modelo generado con la función Cauchit, se tiene que realizar la prueba de líneas paralelas. La Tabla 6-46 muestra el resultado de la prueba.

Tabla 6-46. Prueba de líneas paralelas para UP vs. IU para EI-PREG

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Hipótesis nula	17.581			
General	16.777	0.804	4	0.938

Como se puede observar en el cuadro anterior, el valor de la chi-cuadrado es no significativo y el p-valor es superior a 0.05. Esto significa que el modelo de regresión ordinal calculado cumple con la suposición de rectas paralelas.

En el modelo de regresión lineal, el coeficiente de determinación, R^2 , resume la proporción de la varianza en la variable dependiente asociada con las variables predictoras (variables independientes). Sin embargo, en los modelos de regresión ordinal no se pueden calcular R^2 como en el caso de las regresiones lineales, pero sí se pueden calcular pseudos R^2 . La siguiente tabla muestra los tipos de pseudos R^2 y los valores que toman para el modelo calculado.

Tabla 6-47. Pseudo R^2 calculado para UP vs. IU para EI-PREG

Tipo de Pseudo R^2	Valor
Cox and Snell	0.336
Nagelkerke	0.377
McFadden	0.185

Los resultados que se muestran en la tabla anterior nos indican que el modelo de regresión ordinal explicaría entre el 37.7% y 18.5% de la variabilidad de la utilidad percibida. Es importante resaltar que estos valores tienen que interpretarse con precaución, ya que no son equivalentes directos al R^2 obtenido en un modelo regresión lineal.

El modelo de regresión confirma que la intención de uso es determinada e influenciada por la utilidad percibida. Esto para el caso de la muestra obtenida en la evaluación inicial con los alumnos de pregrado.

6.5.1.2 Modelo de regresión para la relación utilidad percibida-intención de uso para EC3-PREG

Para determinar el modelo de regresión ordinal para la muestra obtenida en el experimento controlado con estudiantes de pregrado EC3-PREG, se utilizó como variable independiente a la intención de uso (IU) y como variable dependiente a la utilidad percibida (UP). La Figura 6-20 muestra la distribución de valores para la variable dependiente IU.

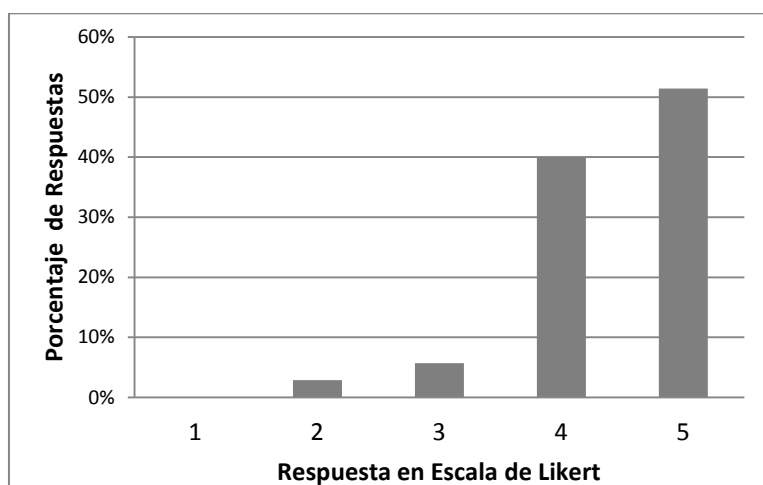


Figura 6-20. Distribución de valores de IU en EC3-PREG

De acuerdo a la información contenida en la Tabla 6-42, la función de enlace más adecuada sería la Log-log complementaria, porque hay muchos valores en las categorías más altas. Con dicha función se obtuvieron los mejores resultados.

Tabla 6-48. Información sobre el ajuste de modelos para UP vs. IU en EC3-PREG

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Solo intersección	24.676			
Final	18.028	6.648	2	0.036

Como se puede observar en la Tabla 6-48, el estadístico chi-cuadrado de la regresión es de 6.648 con 2 grados de libertad, siendo el p-valor de 0.036. Esto indica que el modelo de regresión ordinal obtenido con la función de enlace Log-log complementaria es significativo y provee mejores predicciones que sólo adivinar basándose en las probabilidades marginales de las categorías de respuesta sobre utilidad. También, el modelo obtenido confirma que la utilidad percibida tiene un efecto significativo sobre la intención de usar UCPD para los alumnos que participaron en el estudio EC3-PREG. La Tabla 6-49 muestra el estadístico chi-cuadrado de Pearson y el estadístico chi-cuadrado basado en la desviación. El propósito de estos estadísticos es probar si los datos observados son inconsistentes con el modelo ajustado.

Tabla 6-49. Bondad de ajuste para UP vs. IU en EC3-PREG

	Chi-cuadrado	Gl	p-valor
Pearson	6.894	4	0.142
Desviación	5.299	4	0.258

Como se puede observar en la tabla anterior, los p-valor obtenidos son significativos. Esto quiere decir que los datos y las predicciones del modelo son similares, por lo que se ha obtenido un buen modelo. La Tabla 6-50 muestra la información del modelo de regresión ordinal que se calculó empleando la función Log-log complementaria, que es la función con la se obtuvieron mejores resultados.

Tabla 6-50. Estimación de parámetros para UP vs. IU en EC3-PREG

		Estimación	Error típ.	gl	p-valor
Umbral	[IU = 2]	-4.292	1.067	1	0.000
	[IU = 3]	-3.153	0.688	1	0.000
	[IU = 4]	-1.027	0.409	1	0.012
Ubicación	[UP = 3]	-1.319	0.815	1	0.106
	[UP = 4]	-1.294	0.543	1	0.017
	[UP = 5]	0	.	.	.

Según la tabla anterior y lo indicado por Borooah [Borooah, 2001], para obtener la utilidad percibida (UP) basado en la facilidad de uso percibida (FUP), se necesita calcular lo siguiente: $D = -1.319 \times UP_3 - 1.294 \times UP_4$; donde UP_3 es uno, si la utilidad percibida toma el valor de 3 y 0 en caso contrario; y UP_4 es uno, si la utilidad percibida es 4 y 0 en caso contrario. La UP se determina con el D calculado:

- UP = 2, si $D < -4.292$
- UP = 3, si $-4.292 \leq D < -3.153$
- UP = 4, si $-3.153 \leq D < -1.027$
- UP = 5, si $D \geq -1.027$

Para determinar si se cumple la suposición de que los parámetros de ubicación (coeficientes para las pendientes) son los mismos para todas las categorías de respuesta para el modelo generado con la función Log-log complementaria, se tuvo que realizar la prueba de líneas paralelas. La Tabla 6-51 muestra el resultado de la prueba.

Tabla 6-51. Prueba de líneas paralelas para UP vs. IU en EC3-PREG

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Hipótesis nula	18.028			
General	12.833	5.195	4	0.268

Tal y como se muestra en la tabla anterior, el valor de la chi-cuadrado es no significativo y el p-valor es superior a 0.05. Esto quiere decir que el modelo de regresión ordinal calculado cumple la suposición de rectas paralelas.

De manera similar al modelo de regresión calculado para EI-PREG (ver subsección 6.5.1.1) se calcularon los pseudos R^2 . La siguiente tabla muestra los tipos de pseudos R^2 y los valores que toman para el modelo obtenido.

Tabla 6-52. Pseudo R^2 calculado para UP vs. IU para EC3-PREG

Tipo de Pseudo R^2	Valor
Cox and Snell	0.173
Nagelkerke	0.202
McFadden	0.098

Los resultados que se muestran en la tabla anterior nos indican que el modelo de regresión ordinal explicaría entre el 20.2% y 9.8% de la variabilidad de la utilidad percibida. Es importante resaltar que estos valores tienen que interpretarse con precaución, ya que no son equivalentes directos al R^2 obtenido en un modelo regresión lineal.

El modelo de regresión confirma que la intención de uso es influenciada por la utilidad percibida. Esto para el caso de la muestra obtenida en el experimento controlado con los alumnos de pregrado.

Como se puede observar, el pseudo R^2 obtenido con la muestra del experimento controlado (entre 9.8% y 20.2%) es menor al obtenido con la muestra de la evaluación inicial (entre 18.5% y 37.7%). Esto no contradice a lo obtenido en el análisis de correlación, ya que se obtuvo una relación fuerte con la muestra de la evaluación inicial y una relación moderada con la muestra del experimento controlado. Sin embargo, con los modelos ordinales calculados, se pudo confirmar empíricamente que existe una influencia para la intención de usar UCPD por parte de la utilidad percibida al emplear UCPD en los alumnos de pregrado.

6.5.2 Evaluación con profesionales

La Figura 6-21 muestra el coeficiente de correlación de Spearman, también conocido como rho de Spearman (ρ), entre las variables del MAM para las muestras obtenidas con profesionales. Además, se indica la fortaleza de las relaciones según la Tabla 6-41.

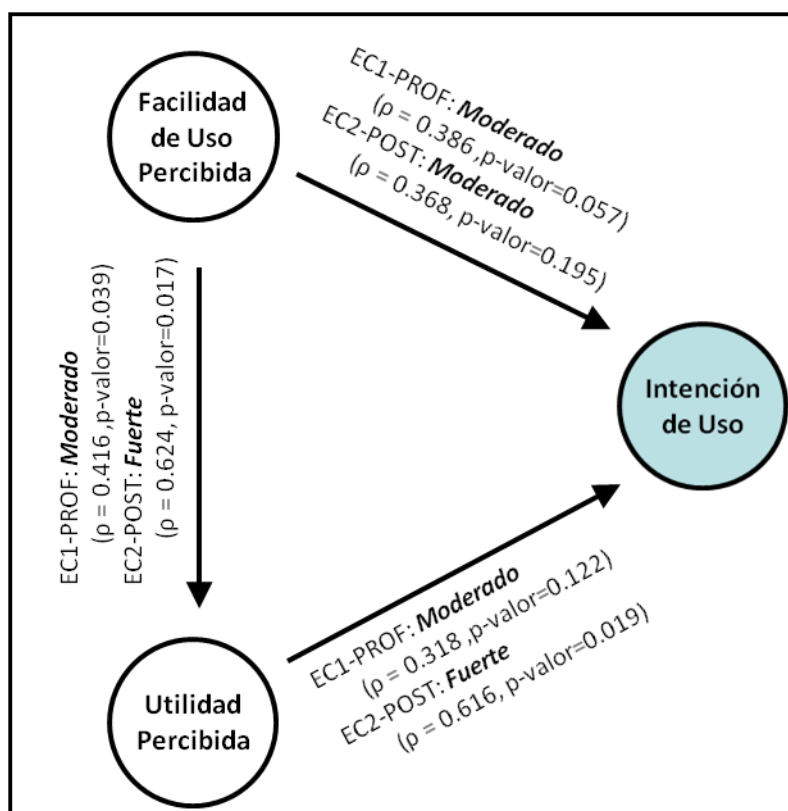


Figura 6-21. Fortaleza de las relaciones del MAM para profesionales

Como se puede observar en la Figura 6-21, las correlaciones entre las variables propuestas por el MAM son moderadas o fuertes para las muestras obtenidas con los profesionales: EC1-PROF y EC2-POST; a diferencia de las muestras con alumnos de pregrado en las que sólo la relación entre utilidad percibida e intención de uso es moderada o fuerte.

Por lo tanto, podemos comprobar empíricamente que existe una dependencia entre las variables facilidad de uso percibida, utilidad percibida e intención de uso, tal y como la propone el MAM. Sin embargo, para confirmar la relación causal entre estas variables, se empleó el análisis de regresión ordinal, el cual se incluye a continuación, de manera similar al utilizado con las muestras obtenidas con los alumnos de pregrado (ver sección 6.5.1).

6.5.2.1 Modelo de regresión para la relación facilidad de uso-utilidad para EC1-PROF.

Para determinar el modelo de regresión ordinal para la muestra obtenida con el experimento controlado con profesionales EC1-PROF, se empleó como variable independiente a la facilidad de uso percibida (FUP) y como variable dependiente a la utilidad percibida (UP). La Figura 6-22 muestra la distribución de valores para la variable dependiente UP.

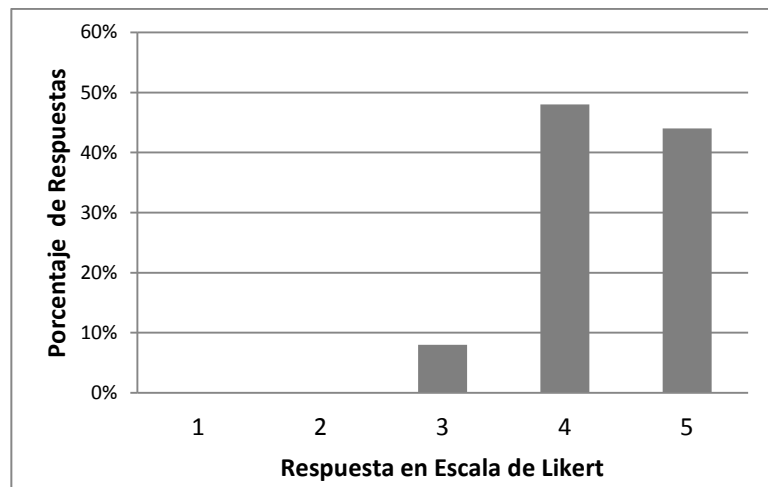


Figura 6-22. Distribución de valores de UP en EC1- PROF

Como los valores son parecidos en las dos últimas categorías (en las dos primeras no se obtuvieron datos), y según la información de la Tabla 6-42, se utilizó la función de enlace Logit.

Tabla 6-53. Información sobre el ajuste de modelos para FUP vs.UP en EC1- PROF

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Solo intersección	19.477			
Final	12.721	6.755	3	0.080

Como se puede observar en la Tabla 6-53, el estadístico Chi-cuadrado de la regresión es de 6.755 con 3 grados de libertad, siendo el p-valor de 0.080. Estos valores indican que el modelo obtenido con la función de enlace Logit provee una mejora significativa que un modelo de sólo intersección. Esto quiere decir que el modelo de regresión ordinal obtenido es significativo y provee mejores predicciones que sólo adivinar basándose en las probabilidades marginales de las categorías de respuesta sobre utilidad. Además, el modelo obtenido confirma que la facilidad de uso percibida tiene un efecto significativo en la utilidad percibida por los profesionales para la muestra EC1-PROF. La Tabla 6-54 muestra el estadístico chi-cuadrado de Pearson y el estadístico chi-cuadrado basado en la desviación.

Tabla 6-54. Bondad de ajuste para FUP vs. UP en EC1-PROF

	Chi-cuadrado	gl	p-valor
Pearson	3.365	3	0.339
Desviación	3.800	3	0.284

Según la tabla anterior, los p-valor obtenidos son significativos. Esto quiere decir que los datos y las predicciones del modelo son similares, por lo que se ha obtenido un buen modelo. La Tabla 6-55 muestra la información del modelo de regresión ordinal obtenido con la función Logit.

Tabla 6-55. Estimación de parámetros para FUP vs. UP en EC1-PROF

		Estimación	Error típ.	gl	p-valor
Umbral	[UP = 3]	-3.998	1.418	1	0.005
	[UP = 4]	-0.436	0.911	1	0.633
Ubicación	[FUP=2]	-2.217	1.864	1	0.234
	[FUP=3]	-3.137	1.571	1	0.046
	[FUP=4]	-0.239	1.065	1	0.822
	[FUP=5]	0	.	0	.

Según la tabla anterior y lo indicado por Borooah [Borooah, 2001], para obtener la utilidad percibida (UP) basado en la facilidad de uso percibida (FUP), se necesita calcular lo siguiente: $D = -2.217 \times FUP_2 - 3.137 \times FUP_3 - 0.239 \times FUP_4$, donde FUP_x toma el valor de uno, si se seleccionó el valor de "x" como facilidad de uso percibida y cero en caso contrario ("x" puede tomar el valor de 2, 3, 4 ó 5). La UP se determina con el D calculado:

- UP = 3, si $D < -3.998$
- UP = 4, si $-3.998 \leq D < -0.436$
- UP = 5, si $D \geq -0.436$

Para determinar si se cumple la suposición de que los parámetros de ubicación (coeficientes para las pendientes) son los mismos para todas las categorías de respuesta para el modelo generado con la función Logit, se tiene

que realizar la prueba de líneas paralelas. La Tabla 6-56 muestra el resultado de la prueba.

Tabla 6-56. Prueba de líneas paralelas para FUP vs. UP en EC1-PROF

Modelo	-2 log de la verosimilitud	Chi-cuadrado	Gl	p-valor
Hipótesis nula	12.721			
General	8.921	3.800	3	0.284

Según lo mostrado en la tabla anterior anterior, el valor de la chi-cuadrado es no significativo y el p-valor es 0.284 el cual es superior a 0.05. Esto significa que el modelo de regresión ordinal calculado cumple con la suposición de rectas paralelas.

En el modelo de regresión lineal, el coeficiente de determinación, R^2 , resume la proporción de la varianza en la variable dependiente asociada con las variables predictoras (variables independientes). Aunque en los modelos de regresión ordinal no se pueden calcular R^2 como en el caso de las regresiones lineales, se pueden determinar pseudos R^2 , los cuales se muestran en la tabla siguiente.

Tabla 6-57. Pseudo R^2 calculado para FUP vs. UP en EC1-PROF

Tipo de Pseudo R^2	Valor
Cox and Snell	0.237
Nagelkerke	0.282
McFadden	0.148

Los resultados que se muestran en la tabla anterior nos indican que el modelo de regresión ordinal explicaría entre el 28.2% y 14.8% de la variabilidad de la utilidad percibida. Hay que tener en cuenta que estos valores tienen que interpretarse con precaución, ya que no son equivalentes directos al R^2 obtenido en un modelo regresión lineal.

El modelo de regresión confirma que la utilidad percibida es determinada e influenciada por la facilidad de uso percibida para la muestra obtenida en el estudio EC1-PROF.

6.5.2.2 Modelo de regresión para la relación facilidad de uso-utilidad para EC2-POST.

Se consideró como variable independiente a la facilidad de uso percibida (FUP) y como variable dependiente a la utilidad percibida (UP), para poder determinar el modelo de regresión ordinal para la muestra obtenida con el experimento controlado con alumnos de posgrado EC2-POST.

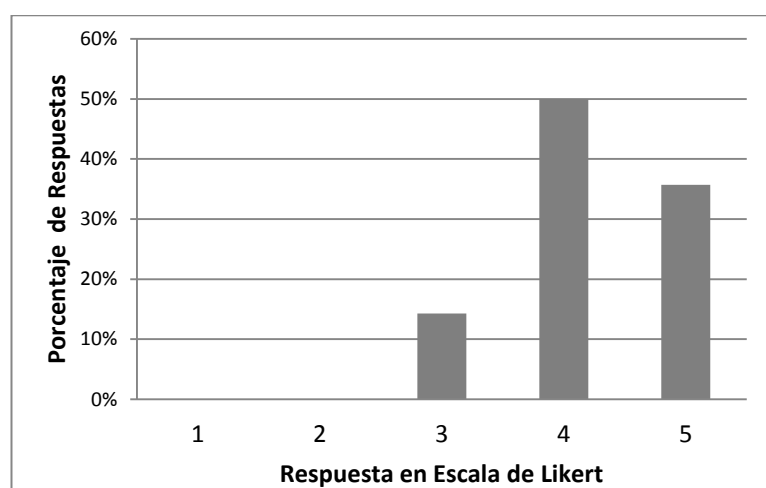


Figura 6-23. Distribución de valores de UP en EC2-POST

Tomando en cuenta la información contenida en la Tabla 6-42, y viendo que la distribución de valores es parecida a la del experimento con profesionales EC1-PROF (ver subsección 6.5.2.1), la función de enlace que se seleccionó fue la Logit.

Tabla 6-58. Información sobre el ajuste de modelos para FUP vs. UP en EC2-POST

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Solo intersección	15.339			
Final	5.778	9.561	2	0.008

La Tabla 6-58 muestra que el estadístico chi-cuadrado de la regresión es de 9.561 con 2 grados de libertad, siendo el p-valor de 0.008. Esto quiere decir que el modelo obtenido con la función de enlace Logit provee una mejora significativa que un modelo de sólo intersección. Por lo tanto, el modelo de

regresión ordinal obtenido es altamente significativo y provee mejores predicciones que sólo adivinar basándose en las probabilidades marginales de las categorías de respuesta sobre utilidad. Además, el modelo obtenido confirma que la facilidad de uso percibida tiene un efecto significativo en la utilidad percibida por los profesionales para la muestra EC2-POST.

Tabla 6-59. Bondad de ajuste para FUP vs. UP en EC2-POST

	Chi-cuadrado	gl	p-valor
Pearson	0.000	2	1.000
Desviación	0.000	2	1.000

La Tabla 6-59, que presenta los estadísticos chi-cuadrado de Pearson y chi-cuadrado basado en la desviación, muestra que los p-valor obtenidos son significativos. Esto nos indica que los datos y las predicciones del modelo son similares, por lo que se ha obtenido un buen modelo. La Tabla 6-60 muestra la información del modelo de regresión ordinal con la función Logit.

Tabla 6-60. Modelo de regresión ordinal para FUP vs. UP en EC2-POST

		Estimación	Error típ.	gl	p-valor
Umbral	[UP = 3]	-22.440	1.225	1	0.000
	[UP = 4]	-0.693	1.225	1	0.571
Ubicación	[FUP=3]	-23.134	0.00	1	.
	[FUP=4]	-1.204	1.426	1	0.398
	[FUP=5]	0	.	0	.

Según la tabla anterior y lo indicado por Borooah [Borooah, 2001], para obtener la utilidad percibida (UP) basado en la facilidad de uso percibida (FUP), se necesita calcular lo siguiente: $D = -23.134 \times FUP_3 - 1.204 \times FUP_4$, donde FUP_x toma el valor de uno, si se seleccionó el valor de “x” como facilidad de uso percibida y cero en caso contrario (“x” puede tomar el valor de 3 ó 4). La UP se determina con el **D** calculado:

- UP = 3, si $D < -22.440$
- UP = 4, si $-22.440 \leq D < -0.693$
- UP = 5, si $D \geq -0.693$

La Tabla 6-61 muestra el resultado de la prueba de líneas paralelas. Esta prueba es importante porque permite determinar si se cumple la suposición de que los parámetros de ubicación (coeficientes para las pendientes) son los mismos para todas las categorías de respuesta para el modelo generado con la función Logit.

Tabla 6-61. Prueba de líneas paralelas para FUP vs. UP para EC2-POST

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Hipótesis nula	5.778			
General	5.778	0.000	2	1.000

En la tabla anterior, se muestra que el valor de la chi-cuadrado es no significativo (0.000) y el p-valor es 1.000 el cual es superior a 0.05. Esto significa que el modelo de regresión ordinal que se ha determinado cumple con la suposición requerida.

Aunque en los modelos de regresión ordinal no se pueden calcular R^2 como en el caso de las regresiones lineales, se pueden determinar pseudos R^2 , los cuales se presentan en la tabla siguiente.

Tabla 6-62. Pseudo R^2 calculado para FUP vs. UP en EC2-POST

Tipo de Pseudo R^2	Valor
Cox and Snell	0.495
Nagelkerke	0.574
McFadden	0.344

Los resultados que se muestran en la tabla anterior nos indican que el modelo de regresión ordinal explicaría entre el 57.4% y 34.4% de la variabilidad de la utilidad percibida. Hay que tener en cuenta que estos valores tienen que interpretarse con precaución, ya que no son equivalentes directos al R^2 obtenido en un modelo de regresión lineal.

El modelo de regresión confirma que la utilidad percibida es determinada e influenciada por la facilidad de uso percibida para la muestra obtenida en el estudio EC2-POST. Adicionalmente, se puede observar que para EC2-POST se obtuvo un mejor modelo que para EC1-PROF, ya que se obtuvieron mejores pseudos R^2 en EC2-POST. En este último, el modelo de regresión explicaría la variabilidad entre las variables entre el 57.4% y 34.4%, a diferencia de EC1-PROF en el que el rango se encuentra entre el 23.7% y 14.8%.

6.5.2.3 Modelo de regresión para la relación facilidad de uso-utilidad vs. intención de uso para EC1-PROF.

Para determinar el modelo de regresión ordinal, se empleó como variable independiente a la intención de uso (IU) y como variables dependientes a la facilidad de uso percibida (FUP) y a la utilidad percibida (UP). La Figura 6-24 muestra la distribución de valores para la variable dependiente IU.

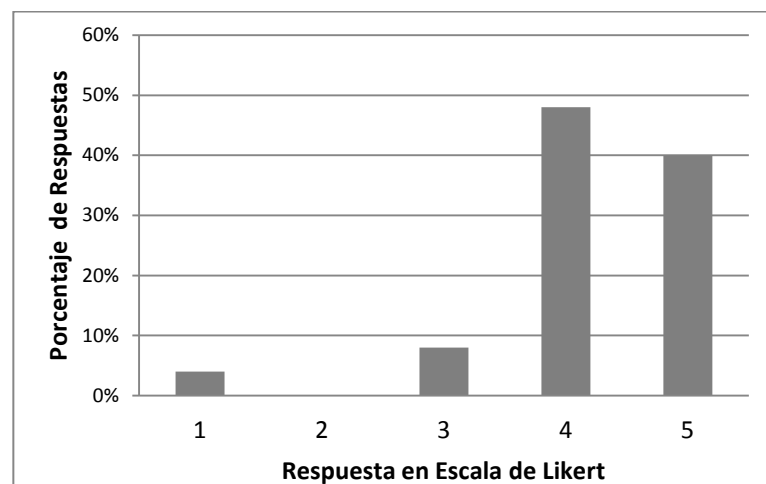


Figura 6-24. Distribución de valores de IU en EC1-PROF

Como se puede observar en la Figura 6-24, la mayoría de los valores de se ubican en las categorías más altas; por ello, y según la información de la Tabla 6-42, la función de enlace más adecuada sería la Log-log complementaria.

Tabla 6-63. Información sobre el ajuste de modelos para FUP-UP vs. IU en E1-PROF

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Solo intersección	30.023			
Final	4.774	28.250	5	0.000

La Tabla 6-63 muestra que el estadístico chi-cuadrado de la regresión es de 28.250 con un p-valor de 0.000, lo que indica que el modelo obtenido con la función de enlace Log-log complementaria provee una mejora significativa en comparación a un modelo de solo intersección. Además, el modelo obtenido confirma que la facilidad de uso percibida y la utilidad percibida tiene un efecto significativo en la intención para usar UCPD en los profesionales, según la muestra de EC1-PROF.

Tabla 6-64. Bondad de ajuste para FUP-UP vs. IU en E1-PROF

	Chi-cuadrado	gl	p-valor
Pearson	4.964	16	0.996
Desviación	6.254	16	0.985

En la Tabla 6-64 que presenta el estadístico chi-cuadrado de Pearson y el estadístico chi-cuadrado basado en la desviación para el modelo, se puede observar que los datos y las predicciones del modelo son similares, por lo que se ha obtenido un buen modelo. La Tabla 6-65 muestra la información del modelo de regresión ordinal que se calculó en SPSS con la función de enlace Log-log complementaria.

Tabla 6-65. Estimaciones de los parámetros para UP- FUP vs IU para EC1-PROF

		Estimación	Error típ.	gl	p-valor
Umbral	[IU = 1]	-23.325	1.479	1	0.000
	[IU = 3]	-21.612	1.130	1	0.000
	[IU = 4]	0.008	0.643	1	0.990
Ubicación	[UP = 3]	-22082	1.769	1	0.000
	[UP = 4]	0.228	0.635	1	0.792
	[UP = 5]	0	.	0	.
	[FUP=2]	-21756	0.000	1	.
	[FUP=3]	-0.267	0.968	1	0.783
	[FUP=4]	0.392	0.729	1	0.591
	[FUP=5]	0	.	0	.

Según la tabla anterior y lo indicado por Borooah [Borooah, 2001], para obtener la intención de uso (IU) basada en la facilidad de uso percibida (FUP) y la utilidad percibida (UP), se necesita calcular lo siguiente: $D = -22.082 \times UP_3 + 0.228 \times UP_4 - 21.756 \times FUP_2 - 0.267 \times FUP_3 + 0.392 \times FUP_4$; donde UP_x toma el valor de 1 si UP es "x" y cero en caso contrario, y FUP_y toma el valor de 1 si FUP es "y" y cero en caso contrario ("x" e "y" pueden tomar valores del 1 al 5). La IU se determina con el D calculado:

- IU = 1, si $D < -23.325$
- IU = 3, si $-23.325 \leq D < -21.612$
- IU = 4, si $-21.612 \leq D < 0.08$
- IU = 5, si $D \geq 0.08$

Para determinar si se cumple la suposición de que los parámetros de ubicación (coeficientes para las pendientes) son los mismos para todas las categorías de respuesta para el modelo generado con la función Log-log complementaria, se tiene que realizar la prueba de líneas paralelas. La Tabla 6-66 muestra el resultado de la prueba.

Tabla 6-66. Prueba de líneas paralelas para FUP – PU vs IU en EC1-PROF

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Hipótesis nula	4.774			
General	0.000	4.774	10	0.906

Como se puede observar en el cuadro anterior, el valor de la chi-cuadrado es no significativo y el p-valor es 0.906, superior a 0.05. Esto significa que el modelo de regresión ordinal calculado cumple con el supuesto requerido. La Tabla 6-67 muestra los tipos de pseudos R^2 y los valores que toman para el modelo calculado.

Tabla 6-67. Pseudo R^2 calculado para FUP – PU vs IU en EC1-PROF

Tipo de Pseudo R^2	Valor
Cox and Snell	0.677
Nagelkerke	0.772
Mcfadden	0.538

Los resultados que se muestran en la tabla anterior nos indican que el modelo de regresión ordinal explicaría entre el 53.8% y 77.2% de la variabilidad de la intención de uso. El modelo de regresión confirma que la intención de uso es determinada e influenciada por la facilidad de uso percibida y la utilidad percibida. Esto para el caso de la muestra obtenida con los profesionales en el estudio EC1-PROF.

6.5.2.4 Modelo de regresión para la relación facilidad de uso-utilidad vs. intención de uso para EC2-POST.

En el caso de la muestra obtenida en el estudio con estudiantes de posgrado EC2-POST, para determinar el modelo de regresión ordinal, se utilizó como variable independiente a la intención de uso (IU) y como variables dependientes a la facilidad de uso percibida (FUP) y a la utilidad percibida (UP).

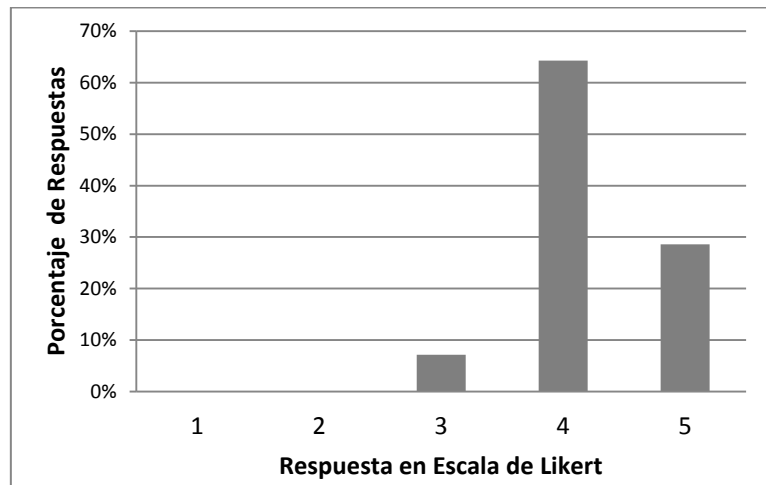


Figura 6-25. Distribución de valores de IU en EC2-POST

La Figura 6-25 muestra una distribución de valores que, sin tomar en cuenta los dos valores más bajos, parecería que tuviera la tendencia a una distribución normal. Por ello, según la información de la Tabla 6-42, **Funciones de enlace por distribución de variable dependiente**, la función de enlace más adecuada sería la Probit. Finalmente, esa función fue la que dio mejores resultados.

Tabla 6-68. Información sobre el ajuste de modelos para FUP-UP vs. IU en EC2-POST

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Solo intersección	15.065			
Final	3.575	11.490	4	0.022

Según la información que se muestra en la Tabla 6-68, el valor del estadístico chi-cuadrado de la regresión es no significativo (el p-valor es menor a 0.022). Esto quiere decir que el modelo obtenido con la función de enlace Probit provee una mejora significativa en comparación a un modelo de solo intersección. Por lo tanto, el modelo confirma que la facilidad de uso percibida y la utilidad percibida tienen un efecto significativo en la intención para usar UCPD en los profesionales (estudiantes de posgrado) según la muestra de EC2-POST.

Tabla 6-69. Bondad de ajuste para FUP-UP vs. IU en EC2-POST

	Chi-cuadrado	gl	p-valor
Pearson	0.103	6	1.000
Desviación	0.168	6	1.000

Según los p-valor obtenidos, con la muestra EC2-POST, para el estadístico chi-cuadrado de Pearson y el estadístico chi-cuadrado basado en la desviación del modelo, se puede concluir que los datos y las predicciones del modelo son similares, por lo que se ha determinado un buen modelo.

La Tabla 6-70 muestra la información del modelo de regresión ordinal que se calculó empleando la función Probit, que es la función con la se obtuvieron mejores resultados.

Tabla 6-70. Estimaciones de los parámetros para UP- FUP vs IU para EC2-POST

		Estimación	Error típ.	gl	p-valor
Umbral	[IU = 3]	-12.410	0.886	1	0.000
	[IU = 4]	0.107	0.823	1	0.897
Ubicación	[UP = 3]	-7.664	.	1	.
	[UP = 4]	-1.405	0.905	1	0.121
	[UP = 5]	0	.	0	.
	[FUP=3]	-4.746	0.000	1	.
	[FUP=4]	0.614	1.015	1	0.545
	[FUP=5]	0	.	0	.

Según la tabla anterior y lo indicado por Borooah [Borooah, 2001], para obtener la intención de uso (IU) basada en la facilidad de uso percibida (FUP) y la utilidad percibida (UP), se necesita calcular lo siguiente: $D = -7.664 \times UP_3 - 1.405 \times UP_4 - 4.746 \times FUP_3 - 0.614 \times FUP_4$; donde UP_x toma el valor de 1 si UP es “x” y cero en caso contrario, y FUP_y toma el valor de 1 si FUP es “y” y cero en caso contrario (“x” e “y” podrían tomar valores del 3 al 5). La IU se determina con el **D** calculado:

- IU = 1, si $D < -12.410$
- IU = 4, si $-12.410 \leq D < 0.107$

- $IU = 5$, si $D \geq 0.107$

A fin de determinar si se cumple la suposición de que los parámetros de ubicación (coeficientes para las pendientes) son los mismos para todas las categorías de respuesta para el modelo generado con la función de enlace Probit, se realizó la prueba de líneas paralelas. La Tabla 6-71 muestra el resultado de la prueba.

Tabla 6-71. Prueba de líneas paralelas para FUP – UP vs IU para EC2-POST

Modelo	-2 log de la verosimilitud	Chi-cuadrado	gl	p-valor
Hipótesis nula	3.575			
General	3.575	0.000	4	1.000

En la tabla anterior, el valor de la chi-cuadrado es no significativo y el p-valor es muy superior a 0.05. Esto significa que el modelo de regresión ordinal pasa la prueba de líneas paralelas y por lo tanto, cumple con el supuesto requerido. La Tabla 6-67 muestra los tipos de pseudos R^2 y los valores que toman para el modelo calculado.

Tabla 6-72. Pseudo R^2 calculado para FUP – UP vs IU en EC2-POST

Tipo de Pseudo R^2	Valor
Cox and Snell	0.560
Nagelkerke	0.691
McFadden	0.494

Los resultados que se muestran en la tabla anterior nos indican que el modelo de regresión ordinal explicaría entre el 69.1% y 49.4% de la variabilidad de la intención de uso. El modelo de regresión confirma que la intención de uso es determinada e influenciada por la facilidad de uso percibida y la utilidad percibida. Esto para el caso de la muestra obtenida con los profesionales en el estudio EC2-POST.

Según los pseudos R^2 que se calcularon, se obtuvo un mejor modelo con la muestra de EC1-PROF que con la muestra de EC2-POST. En el caso de EC1-PROF, el modelo de regresión ordinal que se obtuvo explicaría entre el 77.2% y 53.8% de la variabilidad de la intención de uso; a diferencia del modelo de regresión ordinal obtenido con EC2-POST, el cual explicaría entre el 69.1% y 49.4% de la variabilidad. Sin embargo, se puede observar que para ambas

muestras, EC1-PROF y EC2-POST, se pudo confirmar las relaciones propuestas por el MAM.

6.5.3 Discusión de los resultados

Como se puede observar, los resultados obtenidos con los profesionales comprueban las relaciones entre las variables que se proponen en el MAM. En cambio, esto no sucede con los alumnos de pregrado.

Se puede notar que tanto para los alumnos de pregrado como para los profesionales, la intención de uso es influenciada por la utilidad percibida. Esto no ocurre para las otras relaciones: facilidad de uso vs. utilidad percibida y facilidad de uso vs. intención de uso. En ambas muestras de los alumnos de pregrado EI-PREG y EC3-PREG, la facilidad de uso no influye en la utilidad percibida ni en la intención de uso.

Es imposible decir con certeza el porqué de este fenómeno, pero se puede suponer que los profesionales tienen más experiencia en proyectos de desarrollo de software, por lo que podría asumirse que UCPD es más fácil de usar que las técnicas ad hoc que emplean en sus respectivos trabajos. Además, los alumnos de pregrado no tienen experiencia en emplear técnicas equivalentes a UCPD y esto podría influir en su percepción.

Estos resultados nos pueden sugerir que en alumnos de pregrado no se deberían utilizar cuestionarios relacionados al MAM, o por lo menos preguntas relacionadas a facilidad de uso, para predecir su posible adopción en la práctica. En cambio, sí podríamos utilizar con confianza si se aplican este tipo de cuestionarios a profesionales con experiencia.

6.6 Conclusiones de la Evaluación de UCPD

La evaluación de UCPD se realizó considerando aspectos cuantitativos y cualitativos. Para el aspecto cuantitativo se realizaron tres experimentos controlados, dos de ellos con profesionales y uno de ellos con alumnos de pregrado, en los cuales se evaluó la efectividad de UCPD frente a las técnicas propias que los participantes empleaban para realizar un trabajo equivalente a lo indicado en el experimento controlado.

En cuanto al aspecto cualitativo, se aplicaron cuestionarios a los que emplearon la técnica, en los que tuvieron que indicar, en una escala de 5 niveles, su percepción sobre la facilidad de uso, la utilidad y la intención a usar en un futuro UCPD. Estos cuestionarios fueron aplicados a alumnos de pregrado y a profesionales que tenían al menos dos años de experiencia en proyectos de desarrollo de software.

Los resultados del aspecto cuantitativo muestran que tanto los profesionales como los alumnos de pregrado mostraron mejores resultados que con sus propias técnicas (técnicas ad hoc) para determinar la construcción de

casos de uso, considerando que la secuencia más fácil para desarrollar software es *maestros-transacciones-reportes*.

En cuanto a los resultados de la parte cualitativa, los alumnos de pregrado y los profesionales muestran una actitud positiva frente al empleo de UCPD y consideran que utilizarían la técnica en futuros proyectos de software.

Adicionalmente, con la información de los resultados de la parte cualitativa, se evaluaron las relaciones propuestas por el MAM. Se pudo observar que para los estudiantes de pregrado y para los profesionales la intención de uso es influenciada por la utilidad percibida. También, se pudo comprobar que las relaciones que se encuentran definidas en el MAM se pueden validar empíricamente con los resultados obtenidos con los profesionales, mas no con los resultados obtenidos con los alumnos de pregrado. Esto nos puede sugerir que para predecir la adopción en la práctica, según la propuesta del MAM, se deberían tomar con precaución los resultados que se obtengan al aplicar los cuestionarios a alumnos de pregrado con poca experiencia en proyectos de desarrollo de software.

Aunque se ha empleado modelos de regresión ordinal para determinar de manera empírica las relaciones de causalidad de las variables del MAM, existe una discusión en el entorno científico sobre qué metodo de regresión es el óptimo para datos de tipo ordinal. Por ejemplo, en el campo de la Medicina, Norris et. al [Norris, 2006] realiza un estudio en el que evalúa modelos de regresión ordinal, lineal y logística para datos obtenidos en cuestionario aplicados sobre calidad de vida relacionados a la salud. Los resultados presentados por los autores muestran que los modelos de regresión lineal y ordinal son superiores a los logísticos. Según los resultados de este estudio, se pudo optar por emplear modelos de regresión lineal que son más sencillos de comprender y calcular; sin embargo, se prefirió seguir con las sugerencias de los autores que crearon el modelo de regresión ordinal, el cual fue diseñado para datos de tipo ordinal. Además, existen estudios en los que los resultados obtenidos con los modelos de regresión ordinal fueron satisfactorios [Biesheuvel, 2008] [Chen, 2004]

De acuerdo con los resultados obtenidos con los profesionales, y como trabajo futuro, se deberían evaluar con más detalle las sugerencias de los profesionales que no consideran relevante el mantenimiento de maestros para poder hacer una adaptación de UCPD para este tipo de casos.

También se podría complementar UCPD con otras técnicas que consideran otros puntos de vista diferentes al desarrollador, como es el caso de los usuarios o los clientes del software a construir.

Evaluación de Incremental-FP

Capítulo

7

Este capítulo tiene como finalidad evaluar la técnica Incremental-FP que es incluida en Tupuy para poder estimar el esfuerzo requerido para los incrementos de un proyecto de software.

La evaluación de Incremental-FP se realizó con proyectos de alumnos de pregrado del cuarto año de la carrera de Ingeniería Informática de la Pontificia Universidad Católica del Perú (PUCP). Esta evaluación ha sido realizada de dos maneras: con proyectos en los que los alumnos emplearon Incremental-FP para estimar el esfuerzo requerido para un siguiente incremento o con proyectos en los que los alumnos registraron información para realizar la estimación del esfuerzo (casos de uso construido por incremento y esfuerzo real empleado) y empleando esa información se determinó al final del proyecto si la técnica hubiese dado buenos resultados si se hubiese utilizado. Aunque esta evaluación está relacionada directamente a Incremental-FP, es importante resaltar que los alumnos de pregrado también emplearon UCPD y UML2FP para sus proyectos; por consiguiente, esta evaluación correspondería también a todo Tupuy.

El capítulo se estructura como sigue: la Sección 7.1 presenta una visión general de la evaluación, la Sección 7.2 muestra la información de los proyectos, la Sección 7.3 detalla los resultados obtenidos y, finalmente, se presentan las conclusiones de la evaluación de Incremental-FP.

7.1 Visión General de la Evaluación

Las técnicas definidas en TUPUY han sido empleadas por alumnos de pregrado en proyectos correspondientes a una asignatura del área de Ingeniería de Software desde el año 2004, en algunos semestres de manera completa y en otros parcialmente. UCPD y UML₂FP han sido empleados completamente desde el año 2004; sin embargo Incremental-FP no fue empleado en todos los semestres para determinar el esfuerzo requerido para cada incremento durante todo el proyecto. La Tabla 7-1 muestra por semestre en qué momento y para qué fue empleado Incremental-FP: si fue utilizado para estimar el segundo incremento (segunda columna), si fue empleado para estimar el tercer incremento (tercera columna), si con los datos recopilados durante todo el proyecto fue evaluado conjuntamente con los alumnos para determinar la productividad del equipo al final del proyecto (tercera columna) o si solo se recopilaron los datos, pero los alumnos no emplearon Incremental-FP.

Tabla 7-1. Empleo de Incremental-FP en los proyectos de alumnos por semestre

Semestre	Estima 2do incremento	Estima 3er incremento	Evaluación al final	Sólo se recopilan datos
2004-1	X	X	X	
2004-2	X	X	X	
2005-1	X			
2008-1			X	
2008-2				X

Como se puede observar en la Tabla 7-1, Incremental-FP fue empleada durante todo el proyecto en los semestres 2004-1 y 2004-2. En el semestre 2005-1, fue utilizada para estimar solo el segundo incremento y no se pudieron recopilar los datos para el tercer incremento. En el semestre 2008-1, solo se recopilaron los datos y al final se hizo una evaluación con los alumnos al final del semestre. Finalmente, en el semestre 2008-2, solo se recopilaron los datos del semestre y se procesaron los datos sin los alumnos.

Debido a que los alumnos no tenían información para estimar el primer incremento, se les dio la información obtenida en semestres anteriores para que puedan realizar esta estimación. Sin embargo, no se le dio mucho énfasis a ella, porque la diferencia entre los valores estimados y reales podrían ser muy altos, debido a que esa información no es propia de los equipos. A pesar de ello, sí se les dijo que se repartieran la construcción de casos de uso entre los miembros tomando en cuenta la cantidad de puntos de función que habían calculado de manera uniforme; por ejemplo, un estudiante podía tener asignado un caso de uso de 7 PFSA y otro podría tener asignado dos casos de uso de 3.5 PFSA cada uno.

Utilizando el criterio "PICOC" propuesto por Petticrew y Roberts [Petticrew, 2006] (para este caso no se incluye el criterio de comparación porque no aplica), esta evaluación se definió de la siguiente manera:

Población (*population*): desarrollo de software incremental basado en casos de uso y puntos de función

Intervención (*intervention*): la técnica Incremental-FP

Salidas (*outcomes*): diferencia entre el esfuerzo estimado y real menor al 20%.

Contexto (*context*): estudiantes de pregrado del cuarto año de Ingeniería Informática de la Pontificia Universidad Católica del Perú, quienes realizaron un proyecto de desarrollo de software.

La pregunta de investigación es Incremental-FP permitirá estimar el esfuerzo requerido para un incremento con una desviación menor al 20% si se

compara con el esfuerzo real empleado? Según Hastings et. al [Hastings, 2001], una técnica de estimación se puede emplear con confianza si la diferencia entre el esfuerzo real y el estimado es menor o igual al 20%.

En cuanto a la selección de variables, la variable independiente es la técnica utilizada para realizar la estimación. La variable dependiente es la diferencia entre el esfuerzo estimado y el esfuerzo real empleado para construir el incremento.

7.2 Información de los proyectos

Los proyectos fueron desarrollados por alumnos de pregrado de cuarto año del programa de Ingeniería Informática de la PUCP. La nota del proyecto correspondía al 60% de la nota final de la asignatura, por ello los alumnos le dedican mucho tiempo y esfuerzo al desarrollo del proyecto. En éste, se aplican los conceptos impartidos en las sesiones teóricas de la asignatura.

En esta sección se detalla el conocimiento que tenían los estudiantes al iniciar el proyecto (ver subsección 7.2.1), cómo se distribuyeron los estudiantes en los equipos (ver subsección 7.2.1), las características de los proyectos (ver subsección 7.2.2) y cómo se recolectaron los datos del proyecto (ver subsección 7.2.3).

7.2.1 Conocimiento de los estudiantes y la distribución de los equipos

Los estudiantes fueron divididos de manera aleatoria en equipos de aproximadamente 12 personas. Para la conformación de equipos se siguieron dos criterios: igual proporción de alumnos con rendimiento académico similar entre los equipos e igual proporción de hombres y mujeres. Estos criterios fueron considerados, ya que empíricamente los profesores de la carrera han comprobado que estos factores influyen en el desempeño de los proyectos seguidos en los cursos. La Tabla 7-2 muestra los conocimientos y experiencias de los alumnos antes de comenzar el semestre en el que se realizó esta experimentación.

Tabla 7-2. Conocimientos y experiencias de los estudiantes al inicio del semestre

Característica	Conocimiento y/o Experiencia
Lenguajes y Entornos de Programación	<ul style="list-style-type: none"> • Java, C#, Pascal, C y Prolog. • No conocían Delphi ni Visual Basic .Net
Bases de Datos	<ul style="list-style-type: none"> • Oracle. • No conocían MS SQL Server.
Técnicas de análisis y diseño	<ul style="list-style-type: none"> • Estructurado y orientado a objetos

Característica	Conocimiento y/o Experiencia
Gestión de proyectos	<ul style="list-style-type: none"> Experiencia en proyectos pequeños de programación (equipos de 3 ó 4 estudiantes). No conocían técnicas de planificación ni estimación.

Muchos de los cursos de la carrera de Ingeniería Informática de la PUCP consideran proyectos como parte aplicativa de los conceptos teóricos impartidos. Los alumnos antes de tomar el curso ya tenían experiencia en realizar trabajos grupales, aunque no conocían técnicas de planificación y estimación de proyectos de software.

7.2.2 Características de los proyectos

Cada equipo tenía que desarrollar un sistema de información siguiendo la metodología RUP [IBM, 2006]. La duración del proyecto fue de 14 semanas, que es la duración de un semestre académico de la PUCP. Antes de realizar la fase de construcción, cada equipo tuvo que finalizar el documento de especificación de requisitos (ERS) [IEEE, 1998] con casos de uso. La arquitectura del sistema se basó en el modelo cliente/servidor de dos capas [Sadoski, 1997] y fue validado mediante la realización de un prototipo.

Los temas de los proyectos fueron diferentes en cada semestre, con el fin de evitar posibles copias del trabajo o plagio. Aunque fueron temas diferentes, todos corresponden a un sistema de información de tipo Sistema de procesamiento de transacciones o TPS por sus siglas en inglés (ver capítulo 1 para mayor detalle de los tipos de sistemas de información). En este tipo de sistemas, sus procesos de ingreso y salida de datos son similares; por ello, los resultados entre semestres son comparables. La Tabla 7-2 muestra los temas de los proyectos por semestre.

Tabla 7-3. Temas de proyectos por semestre

Semestre	Tema de proyecto
2004-1	Sistema para una cadena de restaurantes de comida rápida
2004-2	Sistema para una cadena de tiendas por departamento
2005-1	Sistema para una cadena de librerías
2008-1	Sistema para una empresa de distribución y venta de entradas para eventos
2008-2	Sistema para bibliotecas de un centro de educación superior

Todos los equipos tenían que realizar tres iteraciones de la fase de construcción; es decir, tuvieron que construir tres incrementos. La construcción de cada incremento duraba aproximadamente dos semanas. A cada equipo se le asignó un asistente de docencia (al que se le conoce en la PUCP con el nombre

de jefe de práctica) quien se encargaba de asesorar a los estudiantes en el desarrollo del proyecto. Ellos apoyaron a los estudiantes para obtener un ERS que soporte los procesos del negocio del sistema que construirían. Todos los asistentes de docencia trabajan en proyectos de desarrollo de software en la industria con una experiencia promedio de siete años. La Tabla 7-4 muestra el software que utilizaron por semestre los estudiantes.

Tabla 7-4. Software utilizado en los proyectos

Semestre	Herramienta de Desarrollo		Herramienta de Modelado	
	Borland Delphi	Visual Basic . Net	Rational Rose	StarUML
2004-2	X		X	
2005-1	X		X	
2008-1		X		X
2008-2		X		X

En todos los proyectos, los alumnos emplearon como motor de base de datos a Microsoft SQL Server, como sistema operativo a Windows y como herramienta para documentar a Microsoft Office. Las versiones del software empleado, fueron las últimas versiones que existían en el año en que se desarrollaron los proyectos.

7.2.3 Recolección de datos

Semanalmente los alumnos tenían que entregar de manera individual una hoja de cálculo en MS Excel, en el que incluían las horas trabajadas en cada actividad por día. La Figura 7-1 siguiente muestra un ejemplo de la hoja que utilizaron todos los integrantes de los equipos.

		Semana							6	Semana							7
	Actividad	26-09-2004	27-09-2004	28-09-2004	29-09-2004	30-09-2004	01-10-2004	02-10-2004	03-10-2004	04-10-2004	05-10-2004	06-10-2004	07-10-2004	08-10-2004	09-10-2004		
1	Reuniones de coordinación (internas)																
2	Reuniones de control de proyecto (con JP y/o profesor)																
3	Elaboración de informes (no incluye ningún tipo de plan)																
4	Actividades de gestión de proyectos (edt, riesgos, gannt, plan del proyecto, plan de iteración).																
5	Determinación de requisitos (lista de exigencias , especificación casos uso).																
6	Determinación de la arquitectura del sistema																
7	Preparación del plan de pruebas.																
8	Diseño de interfaz de usuario									2	2	2	2	2	2		
9	Diseño (no interfaz de usuario) y programación																
10	Pruebas de software.																
11	Integración de software.																
12	Elaboración de manual de usuario																
TOTAL		0	0	0	0	0	0	0	0	2	2	2	2	2	2		
		Acum. Semanal							0	Acum. Semanal						12	

Figura 7-1. Ejemplo de hoja en Excel para el llenado de horas trabajadas

Es importante mencionar que se les recalcó a los alumnos que la cantidad de horas que utilizaran en el proyecto no iba a influir en la calificación final del curso, que lo importante era que cumplieran con los casos de uso que se habían comprometido a hacer para cada incremento. Esto se hizo para asegurar la honestidad en el registro del esfuerzo realizado por cada uno de los integrantes de cada equipo.

Mediante reuniones con cada equipo, se verificó que el criterio para el llenado de las hojas sea el mismo para todos. Inicialmente hubo problemas, ya que habían alumnos que registraban más horas de lo que realmente utilizaban para el proyecto.

7.3 Resultados Obtenidos.

Solo se procesaron los datos recopilados de aquellos proyectos que concluyeron de manera satisfactoria el software. Se descartaron los datos de aquellos proyectos que fueron considerados poco confiables debido a que se produjeron algunos problemas durante desarrollo del proyecto, entre los que se incluyen discrepancias entre los estudiantes de un equipo. La Tabla 7-5 muestra por semestre y equipo lo siguiente: el esfuerzo real en horas-persona (h-p) y tamaño del incremento en puntos de función sin ajustar (PFSA) para el incremento 1 (columnas 3 y 4), para el incremento 2 (columnas 5 y 6) y para el incremento 3 (columnas 7 y 8). En el caso del incremento 3, se indica con un guión a aquellas celdas en las que no se pudo obtener esa información. En el anexo E se muestra el cálculo de PFSA a mayor detalle.

Tabla 7-5. Esfuerzo real y PFSA para cada incremento

Semestre	Equipo	Incremento 1		Incremento 2		Incremento 3	
		Esfuerzo Real (h-p)	PFSA	Esfuerzo Real (h-p)	PFSA	Esfuerzo Real (h-p)	PFSA
2004-2	A	526	172.66	324	208.22	220	86.12
	B	667	159.13	298	135.87	308	91.00
2005-1	A	248	147.67	238	228.52	-	-
	B	313	149.13	170	141.63	-	-
	C	307	201.94	165	198.41	-	-
2008-1	A	365	116.52	229	130.00	299	166.77
	B	398	107.63	340	173.00	-	-
2008-2	A	205	247.55	256	485.17	-	-
	B	404	197.89	488	472.11	-	-

Como se puede observar en la Tabla 7-5, la cantidad de PFSA por cada proyecto es mayor a 300, este valor no cambiaría mucho si se calculan los puntos de función ajustados. Según Hericko et. al [Hericko, 2008] el repositorio del ISBSG, considera un proyecto pequeño entre 200 y 300 puntos de función. Esto quiere decir que los proyectos realizados por los alumnos no pueden ser considerados como pequeños. No obstante, lo que los alumnos construyen en cada incremento sí se pueden considerar como pequeño, porque la cantidad de PFSA es menor a 300.

Para la fase de construcción, se indicó que la repartición de la carga de trabajo por integrante se debería realizar utilizando los PFSA obtenidos empleando Incremental-FP. Es importante resaltar, que los resultados de la técnica en mención fueron revisados antes que los alumnos lo apliquen en su proyecto, debido a que recién estaban aprendiendo la técnica de Puntos de función.

En Perú, no existen personas certificadas en la técnica de PF, por lo que no se pudieron validar los resultados, aunque sí se siguieron las recomendaciones e indicaciones del manual de PF [Ifpug, 2004].

En los resultados obtenidos en la Tabla 7-5, no se han considerado las horas correspondientes a las reuniones internas de coordinación (punto 1 de la Figura 7-1) y las reuniones con el asistente de docencia o profesor (punto 2 de la Figura 7-1). Esto se debe a que sólo se quería considerar el trabajo efectivo realizado para la construcción del software (diseño, programación y pruebas).

Para los resultados mostrados en las tablas 7 y 8 de esta sección, se ha utilizado la magnitud de error relativo, *Magnitude of Relative Error (MRE)* en inglés, la cual se define en la fórmula 7.1 (y =esfuerzo real, \hat{y} =esfuerzo estimado) [Conte, 1986].

$$MRE = \frac{|y - \hat{y}|}{y} \quad (7.1)$$

Según Hastings et. al [Hastings, 2001], se debe tener en cuenta en relación al MRE que:

- $MRE \leq 20\%$ indica que la técnica puede ser considerada predictiva y puede ser empleada con confianza para la planificación;
- $20\% < MRE \leq 50\%$ indica que es aceptable, pero debe ser empleada con precaución;

- MRE > 50% indica que no es confiable para propósitos de planificación.

Solo se ha incluido la estimación del esfuerzo a partir del segundo incremento, porque la estimación para el primer incremento no sería ni confiable ni precisa. Esto se debe a que se emplearía información de semestres anteriores y esta es información que no es propia de los equipos, por lo que la diferencia entre los valores estimados y reales podría ser alta.

La subsección 7.3.1 muestra los resultados obtenidos al estimar el esfuerzo requerido para el segundo incremento; la subsección 7.3.2 presenta los resultados del tercer incremento; la subsección 7.3.3, la comparación entre los resultados de Incremental-FP y estimaciones globales [Abrahamsson, 2007; y la subsección 7.3.4, la comparación entre Incremental-FP y los puntos de función ajustados que propone la técnica original del Ifpug [Ifpug, 2009].

7.3.1 Segundo incremento

Para estimar el esfuerzo del segundo incremento, se considera que los cambios que se produjeron en el contexto del primer incremento al segundo fue el conocimiento de la herramienta de programación y la experiencia en el desarrollo de aplicaciones. Esto se debe a que los estudiantes nunca habían empleado la herramienta de desarrollo asignada: Borland Delphi del 2004-1 al 2005-1 y Visual Basic .Net en semestres posteriores; por lo que la productividad del primer incremento fue menor a la productividad del segundo incremento. Esta diferencia, empleando Cocomo II, se puede representar mediante el cambio en los multiplicadores de esfuerzo LEXP y AEXP que pertenecen a Cocomo II. En el primer incremento se consideró como “muy baja” en ambos multiplicadores y en el segundo incremento, como “muy alta”. Por lo tanto, el EAF del primer incremento fue de 1.46 y el del segundo incremento, 0.8.

Tomando la fórmula 4.7 del capítulo 4, en el que la productividad estimada para un incremento se calcula en base a la productividad real del incremento previo y el EAF del incremento previo y el del que se va a construir, se pudo calcular la productividad y el esfuerzo estimado del segundo incremento.

La Tabla 7-6 muestra por equipo de cada semestre, en el segundo incremento, la productividad estimada en horas-persona (h-p) por PFSA (tercera columna), esfuerzo estimado en h-p (cuarta columna), el esfuerzo real empleado en el segundo incremento (quinta columna) y la magnitud de error relativo entre el esfuerzo estimado y real del segundo incremento (última columna).

Tabla 7-6. Productividad y esfuerzos estimados para el segundo incremento

Semestre	Equipo	Productividad Estimada (h-p/PFSA)	Esfuerzo Estimado (h-p)	Esfuerzo Real (h-p)	MRE
2004-2	A	1.67	347.58	324.00	7.28%
	B	2.30	312.05	298.00	4.72%
2005-1	A	0.92	210.29	238.00	11.64%
	B	1.15	162.88	170.00	4.19%
	C	0.83	165.28	165.77	0.29%
2008-1	A	1.72	223.14	229.00	2.56%
	B	2.03	350.54	340.00	3.10%
2008-2	A	0.45	220.15	256.00	14.00%
	B	1.12	528.12	488.00	8.22%

Como se puede observar en la Tabla 7-6, las productividades estimadas son diferentes entre equipos, esto quiere decir que no se podría emplear la productividad estimada de un semestre para estimar el esfuerzo de los equipos de otro semestre, incluso entre equipos que correspondan al mismo semestre. En cuanto a las diferencias entre el esfuerzo estimado y el real, el cual se ha medido con el MRE, se puede observar que todas están por debajo del 20%. Esto quiere decir, que según los valores calculados, Incremental-FP puede considerarse como una técnica predictiva que podría emplearse con confianza para la planificación [Hastings, 2001].

7.3.2 Tercer incremento

Para estimar el esfuerzo del tercer incremento, no se consideran que haya cambios en el contexto; por lo que el conocimiento de la herramienta de programación y la experiencia en el desarrollo de aplicaciones es similar al segundo incremento. Esto quiere decir que el EAF del tercer incremento fue el mismo que el del segundo incremento: 0.8.

Tomando la fórmula 4.9 del capítulo 4, en el que la productividad estimada para incrementos a partir del tercero, se calcula en base al promedio de las productividades reales de los incrementos previos con sus EAF, se pudo calcular la productividad y el esfuerzo estimado del tercer incremento.

La Tabla 7-7 muestra por equipo de cada semestre, en el tercer incremento, la productividad estimada en horas-persona (h-p) por PFSA (tercera columna), esfuerzo estimado en h-p (cuarta columna), el esfuerzo real empleado en el tercer incremento (quinta columna) y la magnitud de error relativo entre el esfuerzo estimado y real del tercer incremento (última columna).

Tabla 7-7. Productividad y esfuerzos estimados para el tercer incremento

Semestre	Equipo	Productividad Estimada (h-p/PFSA)	Esfuerzo Estimado (h-p)	Esfuerzo Real (h-p)	MRE
2004-2	A	1.61	138.88	220.00	36.9%
	B	2.25	204.30	308.00	33.7%
2008-1	A	1.74	290.01	299.00	3.0%

De la Tabla 7-7, se puede observar que el MRE es mayor al 33% para el semestre 2004-2 y 3% para el 2008-1, a diferencia de lo obtenido en el segundo incremento en la que todas las estimaciones no pasaron del 14%. Esto podría suponer que la técnica no funcionaría para todos los casos después del segundo incremento.

Inicialmente, para los resultados del 2004-2, se pensó que hubo un error en la recopilación de datos, pero luego de conversar con los alumnos hubo un detalle que no se tomó en cuenta. Para la última entrega, los alumnos tenían que presentar su proyecto a todos sus compañeros del curso, a todos los asistentes de docencia y al profesor, por lo que la presión de presentar un buen trabajo fue mucho mayor que en los otros incrementos. Esto provocó que los equipos hicieran pruebas más intensivas del software e incluso modificaron partes que ya habían sido probadas y que funcionaban correctamente. En el caso del 2008-1 no ocurrió lo mismo, por ello que el MRE fue de 3%.

Para este problema se puede considerar lo que propone Cocomo II [Boehm, 2000a] que incluye un modelo para estimar el porcentaje de tamaño de software que se modificará de las iteraciones anteriores. A diferencia del segundo incremento en la que no hubo problemas en el cálculo, ya que los alumnos no modificaron la funcionalidad hecha en el primer incremento. En la Tabla 7-8 se muestra el porcentaje de modificación realizada de los PFSA del primer y segundo incremento, en el tercer incremento.

Tabla 7-8. Porcentaje de uso de PFSA de 1er y 2do incremento en el 3er incremento

Equipo	PFSA 1er y 2d incremento	PFSA 3er incremento	Total PFSA Usadas 3ra iteración**	% de Uso de PFSA de la 1ra y 2da iteración
A	380.88	86.12	137.79	13.21%
B	295.00	91.00	138.56	15.66%

En la tabla anterior (Tabla 7-8), se puede observar que el porcentaje de modificación realizada de los PFSA de la primera y segunda iteración, en la tercera fue de aproximadamente 14%. Esta información podría ser utilizada en futuras experiencias con alumnos y para una mejora de Incremental-FP.

7.3.3 Comparación de Incremental-FP con estimaciones globales

Según Abrahamsson et. al [Abrahamsson, 2007], los modelos de estimación globales emplean una lista de variables para estimar el esfuerzo de un proyecto completo (tamaño del software, métricas de diseño, factores tecnológicos, etc), se derivan de datos históricos y son empleados para predecir esfuerzo de desarrollo para proyectos similares. En cambio, la principal idea de los modelos de estimación incrementales es que éstos se construyen después de cada iteración, por lo que permiten acomodar cualquier cambio que afecte la estimación del esfuerzo en el transcurso del proyecto de manera más adecuada que en el caso de los modelos de estimación global.

Considerando el concepto de modelo de estimación global e incremental, se pudo realizar una comparación entre un modelo de estimación global genérico e Incremental-FP. Esta comparación se realizó considerando la siguiente:

- los datos de la construcción del primer y segundo incremento, no se emplea la del tercer incremento;
- se toma como esfuerzo estimado del primer incremento al esfuerzo real del primer incremento;
- para estimar el esfuerzo requerido para construir el primer y segundo incremento del modelo de estimación global se emplea la productividad real de la construcción del primer incremento.

La Tabla 7-9 muestra la diferencia entre la estimación global y la estimación empleando Incremental-FP para los dos incrementos. La primera columna de la tabla muestra el semestre al cual corresponde el proyecto; la segunda columna, el equipo; la tercera columna, el esfuerzo real en h-p que tomó construir los dos primeros incrementos; la cuarta columna, el esfuerzo estimado para los dos incrementos siguiendo el modelo global; la quinta columna, el esfuerzo estimado según Incremental-FP; la sexta columna, el MRE del modelo de estimación global; y la última columna, el MRE de Incremental-FP.

Tabla 7-9. Comparación entre Incremental-FP y estimación global para los dos primeros incrementos

Semestre	Equipo	Esfuerzo Real (h-p)	Esfuerzo Estimado Global (h-p)	Esfuerzo Estimado Incremental-FP (h-p)	MRE Global	MRE Incremental-FP
2004-2	A	850.00	1160.33	873.58	36.51%	2.77%
	B	965.00	1236.49	979.05	28.13%	1.46%
2005-1	A	486.00	631.78	458.29	30.00%	5.70%
	B	483.00	610.26	475.88	26.35%	1.47%
	C	472.77	608.64	472.28	28.74%	0.10%

Semestre	Equipo	Esfuerzo Real (h-p)	Esfuerzo Estimado Global (h-p)	Esfuerzo Estimado Incremental-FP (h-p)	MRE Global	MRE Incremental-FP
2008-1	A	594.00	772.23	588.14	30.00%	0.99%
	B	738.00	1037.73	748.54	40.61%	1.43%
2008-2	A	461.00	606.77	425.15	31.62%	7.78%
	B	892.00	1367.82	932.12	53.34%	4.50%

Como se puede observar en la Tabla 7-9, el MRE de Incremental-FP es menor a 7.78% y el del modelo de estimación global es mayor a 28.13%. Esto quiere decir que Incremental-FP daría resultados más precisos en cuanto a la estimación del esfuerzo. Dado que se ha considerado en el primer incremento que el esfuerzo real y el estimado es el mismo, la diferencia entre el valor estimado y el real se debe a la estimación del segundo incremento. Para un mayor análisis, la Tabla 7-10 muestra la comparación entre la estimación del esfuerzo de ambos modelos solo para el segundo incremento. La distribución de la Tabla 7-10 es similar a la de la Tabla 7-9.

Tabla 7-10. Comparación entre Incremental-FP y estimación global para el segundo incremento

Semestre	Equipo	Esfuerzo Real (h-p)	Esfuerzo Estimado Global (h-p)	Esfuerzo Estimado Incremental-FP (h-p)	MRE Global	MRE Incremental-FP
2004-2	A	324.00	634.33	347.58	95.78%	7.28%
	B	298.00	569.49	312.05	91.11%	4.72%
2005-1	A	238.00	383.78	210.29	61.25%	11.64%
	B	170.00	297.26	162.88	74.86%	4.19%
	C	165.77	301.64	165.28	81.97%	0.29%
2008-1	A	229.00	407.23	223.14	77.83%	2.56%
	B	340.00	639.73	350.54	88.16%	3.10%
2008-2	A	256.00	401.77	220.15	56.94%	14.00%
	B	488.00	963.82	528.12	97.50%	8.22%

Como se puede observar en la Tabla 7-10, el MRE del modelo global es mucho mayor que el MRE de Incremental-FP. El MRE del modelo global es mayor al 56.94% y según Hastings et. al [Hastings, 2001], este modelo no sería confiable para propósitos de planificación, si consideramos los resultados de la construcción del segundo incremento.

Según los resultados obtenidos anteriormente, se podría afirmar que Incremental-FP provee mejores resultados para estimar el esfuerzo entre incrementos que un modelo de estimación global genérico.

7.3.4 Comparación de Incremental-FP y estimación con puntos de función ajustados

La técnica de Puntos de función y muchos autores que trabajan con esta técnica sugieren tener información histórica que indique la cantidad de horas-persona o meses-persona requeridas para construir una determinada cantidad de PF. Para ello, se debe emplear los puntos de función ajustados (a estos se les denomina simplemente como “puntos de función”). Para ello, se debe multiplicar a los puntos de función sin ajustar (PFSA) un valor de ajuste que se le denomina *Value Adjustment Factor (VAF)*, factor de valor de ajuste en castellano.

El VAF considera factores como ratio de transacción o reusabilidad para ser calculado. Sin embargo, no considera factores como experiencia en el desarrollo de aplicaciones o experiencia en la herramienta de desarrollo que sí es considerada por Cocomo II y que justo son los factores que cambian entre el primer incremento y el segundo en los proyectos con alumnos. Esto quiere decir que el VAF estimado en todos los proyectos sería el mismo para todos los incrementos. La fórmula 7.2 muestra el cálculo del esfuerzo estimado para el segundo incremento tomando como base la productividad del primer incremento. En la fórmula, se han colocado los puntos de función ajustados como PFSA x VAF

$$EsfuerzoEs\ timado_Incr2 = \frac{EsfuerzoReal_Incr1}{PFsA_Incr1 \times VAF} \times PFsA_Incr2 \times VAF \quad (7.2)$$

En la fórmula 7.2 se puede observar que la productividad del primer incremento se calcula como el esfuerzo real del primer incremento (EsfuerzoReal_Incr1) entre los puntos de función del primer incremento (PFSA_Incr1 x VAF). Para obtener el esfuerzo estimado del segundo incremento (EsfuerzoEstimado_Incr2) se debe multiplicar la productividad con los puntos de función del segundo incremento (PFSA_Incr2 x VAF). Como se puede observar en la fórmula, como el VAF es el mismo, se eliminarían y la fórmula final sería la 7.3.

$$EsfuerzoEs\ timado_Incr2 = \frac{EsfuerzoReal_Incr1}{PFsA_Incr1} \times PFsA_Incr2 \quad (7.3)$$

Empleando la formula 7.3, se realizó la comparación entre la estimación de lo que propone la técnica de Puntos de función e Incremental-FP para el segundo incremento. Esta comparación se muestra en la Tabla 7-11.

Tabla 7-11. Comparación entre Incremental-FP y puntos de función ajustados

Semestre	Equipo	Esfuerzo Real (h-p)	Esfuerzo Estimado Puntos Función Ajustados (h-p)	Esfuerzo Estimado Incremental-FP (h-p)	MRE Puntos Función Ajustados	MRE Incremental-FP
2004-2	A	324.00	634.33	347.58	95.78%	7.28%
	B	298.00	569.49	312.05	91.11%	4.72%
2005-1	A	238.00	383.78	210.29	61.25%	11.64%
	B	170.00	297.26	162.88	74.86%	4.19%
	C	165.77	301.64	165.28	81.97%	0.29%
2008-1	A	229.00	407.23	223.14	77.83%	2.56%
	B	340.00	639.73	350.54	88.16%	3.10%
2008-2	A	256.00	401.77	220.15	56.94%	14.00%
	B	488.00	963.82	528.12	97.50%	8.22%

Como se puede observar en la Tabla 7-11, el MRE de los PFSA es mucho mayor que el MRE de Incremental-FP y según Hastings et. al [Hastings, 2001], los puntos de función ajustados no serían confiables para propósitos de planificación, si consideramos los resultados de la construcción del segundo incremento. Además, podemos afirmar que el VAF no refleja los cambios de contexto de un proyecto, por lo que su uso es innecesario para la planificación de un proyecto que siga un modelo de ciclo de vida incremental.

Si se compara la Tabla 7-11 con la Tabla 7-10 que se incluye en la subsección 7.3.3 (subsección anterior), se podrá notar que son iguales. Esto se debe a que la fórmula 7.3 que se obtuvo es la misma que la del modelo de estimación global.

7.4 Conclusiones de la Evaluación de Incremental-FP

Aunque la información obtenida corresponde a nueve proyectos y los resultados obtenidos son alentadores para la técnica propuesta, aún no se puede afirmar que la técnica sea realmente confiable para otros contextos. Algunos de los comentarios de los alumnos con respecto a la técnica fueron los siguientes:

- la técnica les sirvió para medir el tamaño de la parte del software que tenían que implementar cada integrante para una iteración;
- el esfuerzo registrado en la primera iteración fue muy útil para calcular el esfuerzo de la segunda iteración.

También, se puede observar en la Tabla 7-12 que la productividad real entre los equipos es diferente. Por ejemplo, para los equipos del semestre en el segundo incremento, el equipo A requirió de 1.56 horas-persona/PFSA, en cambio el equipo B requirió de 2.19 horas-persona/PFSA. Algo similar ocurre para los otros incrementos.

Tabla 7-12. Productividad real para los proyectos del semestre 2004-2

Equipo	Productividad Real 1er Incremento (p-h/PFSA)	Productividad Real 2do Incremento (p-h/PFSA)
A (11 alumnos)	3.05	1.56
B (12 alumnos)	4.19	2.19

La variación que se muestra en la Tabla 7-12 se debe a muchos factores, alguno de ellos son: la cantidad de alumnos por equipo y la capacidad de trabajo en equipo, siendo algunos de ellos difíciles de cuantificar. Lo que sí se pudo comprobar es que la información del esfuerzo real utilizado es muy útil para la estimación del esfuerzo del equipo a quien le pertenecen esos datos.

En cuanto a los resultados obtenidos en la tercera iteración, para el semestre 2004-2, el MRE fue aproximadamente al 35% para todos los proyectos y equipos (ver Tabla 7-7). Como se comentó, esto se debía a los cambios que realizaron los primeros incrementos, debido a la presión producida por la presentación final. Además, se pudo observar que el porcentaje de modificación realizada para la construcción del primer y segundo incremento en el tercero fue de aproximadamente 14%, siendo el porcentaje para ambas técnicas muy similar. Esta información será utilizada para mejorar Incremental-FP.

Es difícil hacer generalizaciones de los resultados obtenidos para otros contextos que no sea el académico. La ventaja de realizar experimentaciones con alumnos es el que se puedan controlar algunos factores o variables que no se pueden controlar en proyectos en la industria. Algunos de ellos son los siguientes:

- Conocimiento y experiencia de los miembros de los equipos de desarrollo

En proyectos de la industria es muy difícil encontrar que la experiencia de todos los miembros sea la misma. En algunos casos, los miembros no tienen los conocimientos necesarios sobre desarrollo orientado a objetos.

- Rotación de personal

En la industria es bastante frecuente el cambio de los integrantes de un proyecto de software por diversos factores, entre ellos, la asignación a otros proyectos. Para la experimentación, este factor no se produjo.

- Cambio de requisitos

El cambio de requisitos es algo que ocurre frecuentemente en la industria. Este factor se puede controlar en un proyecto académico.

- Secuencia de implementación de los requisitos

En esta experimentación se ha realizado la implementación de los requisitos siguiendo el diagrama de precedencias, lo cual a veces no se puede hacer realizar en la industria. Esto se debe a que la precedencia de construcción de los requisitos depende de los requerimientos de los usuarios y de los clientes.

Los resultados obtenidos se podrían aplicar en la industria cuando el equipo de trabajo cuente mucha experiencia en el desarrollo de aplicaciones y tenga que afrontar un proyecto en el que se tenga que utilizar una nueva herramienta y lenguaje de programación.

Incremental-FP propone fórmulas simples que no incluyen exponentes, como sí lo indica Cocomo II. Los resultados obtenidos en los proyectos con estudiantes nos sugieren que no era necesario emplear exponentes en dichas fórmulas. El manual del modelo de Cocomo II [Boehm, 2000b] señala que el exponente para el tamaño del software debe ser menor a uno para proyectos pequeños, pero los proyectos de los alumnos no pueden ser considerados como pequeños según el ISBSG [Hericko, 2008].

Finalmente, podemos afirmar que Incremental-FP provee mejores resultados que un modelo de estimación global genérico. También, provee mejores resultados que emplear los puntos de función ajustados que sugiere el Ifpug [Ifpug, 2009].

Conclusiones y Líneas de Trabajo a Futuro

Capítulo

8

Tras exponer la solución propuesta y los resultados obtenidos en las evaluaciones de la misma, en este capítulo se detallan las conclusiones derivadas del presente trabajo de investigación. Asimismo se incluyen las líneas de trabajo futuras que se podrían desprender y que continuarían el trabajo realizado.

El capítulo se ha dividido en las siguientes secciones: la Sección 8.1 muestra las conclusiones del conjunto de técnicas a la que se le ha denominado Tupuy; la Sección 8.2 detalla las conclusiones y líneas de trabajo futuro de UML₂FP; la Sección 8.3 presenta las conclusiones y líneas de trabajo futuro de UCPD, la Sección 8.4 muestra las conclusiones y líneas de trabajo futuro de Incremental-FP y, finalmente, la Sección 8.5 indica la publicación de resultados.

8.1 Tupuy

La estimación del esfuerzo es muy diferente para un proyecto que sigue un ciclo en cascada que para uno que emplee un modelo de ciclo de vida incremental. La mayoría de técnicas existentes para realizar la estimación del esfuerzo en proyectos software, como Puntos de función, fueron pensadas para realizar esta tarea para proyectos que siguen un modelo de ciclo de vida en cascada de Royce [Royce, 1970].

Es por ello que esta investigación aborda el problema de definir una aproximación que permita apoyar la estimación y planificación de proyectos de desarrollo de software empleando modelos ciclos de vida incremental, un paradigma orientado a objetos y que utilicen la técnica de Puntos de función.

Como se ha detallado en el capítulo 2, el estudio del estado de la cuestión ha permitido observar que no existe una técnica que cubra todas las características del problema que aborda la solución que se está proponiendo. Si bien existen algunas técnicas que permitan realizar el cálculo de Puntos de Función para modelos orientado a objetos, no existen técnicas que permitan adaptar los PF a modelos de ciclo de vida incremental. En cuanto a la priorización o definición de la secuencia de construcción de casos de uso, las técnicas toman en cuenta mayormente el punto de vista del usuario o del cliente, pero muy poco la facilidad para construir software desde la perspectiva del desarrollador.

Con el objetivo de ofrecer una solución que cumpla con la problemática encontrada, este trabajo propone el empleo de un conjunto de técnicas que

permitan apoyar la estimación y planificación de proyectos de software que empleen modelos ciclos de vida incremental, paradigma orientado a objetos y Puntos de función. A este conjunto de técnicas se le ha denominado Tupuy, la cual está conformada por las técnicas UML₂FP, UCPD e Incremental-FP. La evaluación de estas técnicas se ha realizado mediante experimentos controlados con estudiantes de pregrado, estudiantes de posgrado y profesionales. También, Tupuy ha sido utilizada en proyectos realizados por alumnos de pregrado con resultados alentadores.

Tupuy se basa en los modelos de ciclo de vida incremental, los Puntos de función y la priorización de la construcción de incrementos de software tomando en cuenta el punto de vista del desarrollador. Por ello, el trabajo a futuro que se desprendería de esta investigación correspondería a adaptar Tupuy para modelos de ciclo de vida diferentes al incremental (por ejemplo: iterativo o evolutivo), técnicas diferentes a Puntos de función (por ejemplo: Cosmic o Puntos de casos de uso) o complementar la priorización de la construcción de incrementos de software con propuestas de otras técnicas que toman en cuenta el punto de vista del resto de *stakeholders* de un proyecto de software, por ejemplo el punto de vista de los clientes.

Como las técnicas que propone Tupuy se complementan entre sí, para una mayor comprensión sobre el aporte que realiza cada una de ellas y qué líneas de trabajo futuro se desprenderían relacionadas a cada una de ellas, se han detallado las conclusiones y trabajo futuro por cada técnica. Por ello, se ha creado una sección para UML₂FP (Sección 8.2), una para UCPD (Sección 8.3) y, finalmente, una para Incremental-FP (Sección 8.4) que detallan lo indicado.

8.2 UML₂FP

UML₂FP es una técnica que permite calcular los Puntos de función, empleando para ello modelos orientados a objetos. Además, esta técnica cumple con las especificaciones de los dos últimos manuales de Puntos de función [Ifpug, 2004] [Ifpug, 2009].

Para el cálculo de los PF de los ficheros, UML₂FP especifica reglas que consideran las relaciones que podrían existir entre las clases de la fase de análisis: asociación, agregación, composición, herencia y clase asociación. Para definir estas reglas, se tomó como base los patrones propuestos por Coad [Coad, 1997]. Según los resultados obtenidos con los experimentos controlados con profesionales y estudiantes de pregrado, se pudo probar empíricamente que las reglas definidas de UML₂FP para determinar los ficheros y sus RET proveen resultados por lo menos igual de precisos que la técnica original propuesta por el Ifpug. En algunos experimentos controlados, las reglas relacionadas a composición que define UML₂FP permitieron que los resultados fueran mejores que con la técnica original del Ifpug.

Tupux [Balbin, 2009] es un plug-in de StarUML que automatiza Tupuy y que genera automáticamente los ficheros con sus RET y DET a partir de un diagrama

de clases, siguiendo las reglas definidas por UML₂FP. Esto facilitaría realizar esta tarea y reduciría los errores de aplicar una regla de manera inadecuada, errores que fueron detectados al revisar las fichas que fueron llenadas por los participantes de los experimentos controlados. Los resultados obtenidos preliminarmente con esta herramienta son satisfactorios, pero se tendría que realizar más evaluaciones para determinar su aplicabilidad en proyectos reales.

En cuanto a las transacciones de PF, UML₂FP propone utilizar el concepto de *Use Case Realizations*, realización o implementación de casos de uso en castellano, de RUP [IBM, 2006] y colaboración de UML [OMG, 2008] para representar dichas transacciones. Este elemento se podría considerarse el más independiente entre las diferentes metodologías de desarrollo de software. Además, no añade ningún diagrama adicional que solo pueda ser empleado para el cálculo de Puntos de función y no sea útil para etapas como el diseño.

El cálculo de transacciones puede ser apoyado por herramientas integradas a modeladores de UML como Tupux. Si se emplean este tipo de herramientas, ya no sería necesaria la creación de diagramas de UML que solo sean útiles para el cálculo de Puntos de función como las propuestas por [Uemura, 2001] [Lavazza, 2008].

Relacionado al cálculo de Puntos de función para modelos orientados a objetos se pueden aún desprender líneas de trabajo a futuro. Una de ellas, está relacionada con determinar si la técnica de Puntos de función podría simplificarse. Por ejemplo, el concepto de dependencia entre clases o entidades para determinar si son dos ficheros o un fichero con dos RET, debería ser evaluada; porque a lo mejor no es necesaria y finalmente la estimación no variará si solo se empleara cada clase como si fuera un fichero.

Aunque no existe una técnica específica para determinar formalmente que es un *Use Case Realization*, por los ejemplos revisados de RUP, se observó que el concepto de transacción de Puntos de función podría ayudar en determinar *Use Case Realization* de manera más formal para sistemas de información del tipo de procesamiento de transacciones. Incluso, el método de Jaaksi [Jaaksi, 1999] podría ser mejorado con el concepto propuesto por Puntos de función, porque el autor propone definir una lista de operaciones, en paralelo a la realización de un diagrama de clases de análisis, pero no tiene guías claras para definir que es una operación. Para este método, como una mejora, se podría definir que una operación es una transacción de Puntos de función.

En cuanto al método de Larman [Larman, 1999], que propone el uso de un diagrama de secuencias del sistema para determinar las operaciones que los actores realizan con el sistema, se podría evaluar si es más simple y fácil el utilizar el concepto de transacciones de PF para determinar operaciones que realizar un diagrama de secuencias del sistema.

Finalmente, las preguntas de investigación para un trabajo futuro relacionado UML₂FP son las siguientes:

- ¿La eliminación del concepto de dependencia entre clases para determinar ficheros y RET permitirá obtener una estimación del esfuerzo tan igual que si se considerara estos conceptos?
- ¿El concepto de transacciones podría apoyar en la definición de *Use Case Realization* de la metodología RUP?
- ¿El concepto de transacciones podría apoyar en la definición de las operaciones del método propuesto por Jaaksi?
- ¿El concepto de transacciones podría ser más fácil y simple de aplicar que realizar el diagrama de secuencias del sistema de Larman para definir las operaciones de un sistema?
- ¿Será aplicable UML₂FP en proyectos de desarrollo de software en la industria?

8.3 Use Case Precedence Diagram

Use Case Precedence Diagram (UCPD) es una técnica que permite priorizar los casos de uso teniendo en cuenta la facilidad de construcción de software desde el punto de vista del desarrollador. Para ello, considera el concepto de tablas maestras y tablas de transacciones de un sistema de información, que es muy conocida entre la comunidad de desarrolladores de software. También, toma en cuenta las precondiciones de los casos de uso.

UCPD, adicionalmente a las relaciones entre casos de uso que especifica UML (*include*, *extend* y *generalization*), propone la inclusión de una nueva relación: *antecede*. La idea de este diagrama fue tomado de Doug Rosenberg [Rosenberg, 1999] quien propone el uso de un diagrama parecido que especifica las relaciones *precede* e *invoke* entre casos de uso.

Los resultados de los experimentos controlados con profesionales y alumnos de pregrado permitieron corroborar que la efectividad de UCPD es mayor frente a las técnicas propias que los participantes empleaban para realizar un trabajo equivalente a lo indicado en el experimento controlado.

Adicionalmente, se aplicaron cuestionarios a los que emplearon la técnica, en los que tuvieron que indicar, en una escala de cinco niveles, su percepción sobre la facilidad de uso, la utilidad y la intención a usar en un futuro UCPD. Los resultados nos permitieron comprobar que los alumnos de pregrado y los profesionales muestran una actitud positiva frente al empleo de UCPD y consideran que utilizarían la técnica en futuros proyectos de software.

Con la información de los cuestionarios que permitieron observar la percepción que tuvieron los participantes a los experimentos sobre UCPD, se evaluaron las relaciones propuestas por el MAM. Se pudo comprobar que las relaciones que se encuentran definidas en el MAM se pueden validar

empíricamente con los resultados obtenidos con los profesionales, mas no con los resultados obtenidos con los alumnos de pregrado. Esto nos puede sugerir que para predecir la adopción en la práctica, según la propuesta del MAM, se deberían tomar con precaución los resultados que se obtengan al aplicar los cuestionarios a alumnos de pregrado con poca experiencia en proyectos de desarrollo de software.

De acuerdo con los resultados obtenidos con los profesionales, y como trabajo futuro, se deberían evaluar con más detalle las sugerencias de algunos profesionales que no consideran relevante el mantenimiento de tablas maestras para poder hacer una adaptación de UCPD para este tipo de casos.

También se podría complementar UCPD con otras técnicas que consideran otros puntos de vista diferentes al desarrollador, como es el caso de los usuarios o los clientes del software a construir.

Por las consultas realizadas a desarrolladores que emplean enfoques ágiles, UCPD es perfectamente compatible a estos enfoques. Sin embargo, faltarían realizar experimentos controlados con desarrolladores expertos en metodologías ágiles para determinar su aplicabilidad de UCPD en proyectos que utilizan este tipo de metodologías.

Finalmente, las preguntas de investigación para un trabajo futuro relacionado UCPD son las siguientes:

- ¿Es posible adaptar UCPD para que se incluyan las sugerencias de algunos profesionales que consideran que no son relevantes el mantenimiento de tablas maestras para definir la secuencia de construcción de casos de uso?
- ¿Es posible complementar UCPD con otras técnicas que consideran los puntos de vista de los usuarios y de los clientes?
- ¿Es aplicable el empleo de UCPD en proyectos de desarrollo de software que utilizan metodologías ágiles?

8.4 Incremental-FP

Incremental-FP es una técnica que, después de determinar la secuencia de construcción de casos de uso con UCPD y de realizar el cálculo de Puntos de función sin ajustar empleando UML₂FP, permite determinar el esfuerzo que tomará la construcción de cada incremento en aquellos proyectos de software que siguen un modelo de ciclo de vida incremental.

La primera actividad de Incremental-FP consiste en determinar los PFSA para cada incremento; la segunda actividad, en definir qué caso de uso se construirá en cada incremento y el esfuerzo que se requiere para construirlos; y la tercera actividad, en evaluar y revisar lo ocurrido en la construcción de los incrementos para realizar ajustes necesarios. Además, Incremental-FP no

emplea exponentes en sus fórmulas de estimación como sí lo propone otros modelos de estimación como Cocomo II.

Los resultados obtenidos con Incremental-FP en nueve proyectos con alumnos de pregrado de cuarto año de la carrera de Ingeniería Informática de la PUCP son alentadores. Los proyectos de los alumnos, según el ISBSG no pueden ser considerados como pequeños.

Los alumnos comentaron que la técnica les ayudó a estimar el tamaño del software a desarrollar y qué fue útil para calcular el esfuerzo necesario para construir cada incremento. También, se pudo confirmar que la productividad entre equipos es diferente, por lo que no sería adecuado realizar la estimación del esfuerzo de un proyecto si no se utiliza la información del propio equipo de desarrollo.

Al comparar Incremental-FP con otros modelos de estimación, se pudo comprobar que Incremental-FP provee mejores resultados que un modelo de estimación global genérico. También, provee mejores resultados que emplear los puntos de función ajustados que sugiere el Ifpug.

Sin embargo, aún no se puede afirmar que la técnica sea realmente confiable para otros contextos, como en la industria; porque durante los proyectos con alumnos se pudo controlar factores que pueden afectar la estimación del esfuerzo como rotación de personal, cambio de requisitos, secuencia de construcción de requisitos diferente a la propuesta por UCPD, etc.

Según los resultados obtenidos en el tercer incremento del semestre 2004-2, en el que el MRE fue mayor al 30% para todos los proyectos y equipos (ver Tabla 7-7), el porcentaje de modificación realizada para la construcción del primer y segundo incremento en el tercero fue de aproximadamente 14%. Como se puede observar, los alumnos en este caso emplearon un modelo de ciclo de vida iterativo y no incremental al final del proyecto. Esto quiere decir, que Incremental-FP podría considerar estos datos para realizar una propuesta para modelos de ciclo de vida iterativo.

Las metodologías ágiles se basan en el desarrollo de software incremental. Por ello, Incremental-FP podría ser utilizado con este tipo de metodologías, pero se tendría que realizar una evaluación con desarrolladores expertos en el tema para determinar si es aplicable o no a estas metodologías. También, podría adaptarse Incremental-FP para otras técnicas de estimación como Cosmic o Puntos de casos de uso.

Finalmente, las preguntas de investigación para un trabajo futuro relacionado Incremental-FP son las siguientes:

- ¿Es aplicable el empleo de Incremental-FP en proyectos en la industria?
- ¿Se podría adaptar Incremental-FP para proyectos que sigan un modelo de ciclo de vida iterativo?

- ¿Es aplicable el empleo de Incremental-FP en proyectos de desarrollo de software que utilizan metodologías ágiles?
- ¿Es posible adaptar lo que propone Incremental-FP para otras técnicas de estimación como Cosmic o Puntos de casos de uso?

8.5 Publicación de Resultados

Los resultados alcanzados en esta investigación fueron presentados en eventos, libros y revistas internacionales relacionados al tema de la tesis, con comité de revisión.

Sobre **UCPD**, las publicaciones fueron las siguientes:

1. Pow-Sang, J.A., Moreno, A.M, Imbert, R., A Replicated Experiment with Undergraduate Students to Evaluate the Applicability of a Use Case Precedence Diagram Based Approach in Software Projects, Software Engineering, Business Continuity, and Education .Communications in Computer and Information Science, Springer-Verlag, 2011.
2. Pow-Sang, J.A., A replicated experiment to evaluate the applicability of a use case precedence diagram-based approach in software development project, Libro: "Software Engineering: Methods, Modeling and Teaching", Universidad de Medellín, 2011.
3. Pow-Sang, J.A. Evaluating the Intention to Use the UCPD in Software Development Projects, Proceedings FGIT 2010, CCIS, Lecture Notes in Computer Science, Springer-Verlag,2010.
4. Pow-Sang, J.A., Nakasone, A., Moreno, A.M, Imbert, R., Evaluating the Applicability of a Use Case Precedence Diagram based Approach in Software Development Projects through a Controlled Experiment, Advances in Security Technology, Revised Selected Papers of SecTech 2008, Communications in Computer and Information Science (CCIS), Lecture Notes in Computer Science, Springer-Verlag, 2009.
5. Pow-Sang, J.A., Nakasone, A., Imbert, R., Moreno, A.M., An Approach to Determine Software Requirement Construction Sequences based on Use Cases, Proceedings Advanced Software Engineering and Its Applications-ASEA 2008 (Sanya, China), IEEE Computer Society, 2008,

Sobre **UML₂FP**, las publicaciones fueron las siguientes:

6. Pow-Sang, J.A., Nakasone, A., Gasco, L. Evaluating the Applicability of a Function Point Logic File. Identification Technique through Controlled Experiments, International Journal of Software Engineering and Its Applications, Vol 4, Num 3, 2010, SERSC, 2010.

7. Pow-Sang, J.A., Gasco, L., Nakasone, A., A Function Point Logic File Identification Technique Using UML Analysis Class Diagrams, Proceedings ASEA 2009, CCIS, Lecture Notes in Computer Science, Springer-Verlag, 2009.
8. Pow-Sang, J.A., Nakasone, A., Imbert, R., Moreno, A.M., Case Study Evaluations for a Function Point Counting Improvement for Object-Oriented Projects, Mexican International Conference on Computer Science 2008, Mexico, 2008.
9. Pow-Sang, J.A., Imbert, R., Including the Composition Relationship among Classes to Improve Function Points Analysis, JPC'07, Trujillo, Peru, 2007.

Sobre **Incremental-FP**, las publicaciones fueron las siguientes:

10. Pow-Sang, J.A., Jolay-Vasquez, E., An Approach of a Technique for Effort Estimation of Iterations in Software Projects, Proceedings XIII Asia Pacific Software Engineering Conference (Bangalore-India), IEEE Computer Society, USA, 2006.
11. Pow-Sang, J.A., Una Experiencia con Estudiantes para la Estimación del Esfuerzo de cada Iteración en Proyectos de Software, Proceedings XXXI Conferencia Latinoamericana de Informática-CLEI 2005, Cali-Colombia, 2005.
12. Pow-Sang, J.A., Estudio Comparativo de Técnicas para la Estimación del Esfuerzo de las Iteraciones de Proyectos Software, Proceedings 4tas Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento-IIISIC'04, Madrid-España, 2004.
13. Pow-Sang, J.A., Imbert R., Estimación y Planificación de Proyectos Software con Ciclo de Vida Iterativo-Incremental y empleo de Casos de Uso, Proceedings VII Workshop Iberoamericano de Ingeniería de Requisitos y Desarrollo de Ambientes de Software- IDEAS 2004, Arequipa-Perú, 2004.
14. Pow-Sang, J.A., Estudio de Técnicas Basadas en Puntos de Función para la Estimación del Esfuerzo en Proyectos de Software, Revista Sistemas e Informática, Universidad Nacional Mayor de San Marcos, Lima-Perú, 2004.

Sobre **Tupux**, las publicaciones fueron las siguientes:

15. Balbin, D., Ocrosopoma, M., Soto, E., Pow-Sang, J.A., TUPUX: An Estimation Tool for Incremental Software Development Projects, Proceedings AST 2009, IEEE Computer Society, 2009.

Bibliografía

[Abrahao, 2004] Abrahao S., On the Functional Size Measurement of Object-Oriented Conceptual Schemas: Design and Evaluation Issues, Tesis Doctoral, Universidad Politécnica de Valencia, 2004.

[Abrahao, 2006] Abrahao S., Poels G., Pastor O., A functional size measurement method for object-oriented conceptual schemas: Design and evaluation issues, Software and Systems Modeling, Vol. 5, No. 1, pp. 48-71, Springer, Alemania, 2006.

[Abrahao, 2007] Abrahao S., Poels G., Experimental evaluation of an object-oriented function point measurement procedure, Information and Software Technology Vol. 49, No. 4, pp 366-380, Elsevier, 2007.

[Abrahamsson, 2007] Abrahamsson, P., Moser, R., Pedrycz, W., Sillitti, A., Succi, G., Effort Prediction in Iterative Software Development Processes -- Incremental Versus Global Prediction Models, Proceedings ESEM 2007, pp.344-353, IEEE Computer Society, EEUU, 2007.

[Albrecht, 1979] Albrecht, A. J. Measuring Application Development Productivity, IBM Applications Development Symposium, Monterey, CA, EEUU, 1979.

[Ambler, 2004] Ambler, S., The Object Primer: Agile Model-Driven Development with UML 2.0, Cambridge University Press, Inglaterra, 2004.

[Antoniol, 1999] Antoniol G., Lokan C., Caldiera G., Fiutem R., Function point-like measure for object-oriented software, Empirical Software Engineering, Vol. 4, No. 3, pp. 263-287, Springer, Alemania, 1999.

[Antoniol, 2003] Antoniol G., Fiutem R., Lokan C., Object-oriented function points: An empirical validation, Empirical Software Engineering, Springer, Alemania, Vol. 8, No. 3, pp. 225-254, Springer, Alemania, 2003.

[Balbin, 2009] Balbin, D., Ocrosopoma, M., Soto, E., Pow-Sang, J.A., TUPUX: An Estimation Tool for Incremental Software Development Projects, Proceedings AST'09, pp. 39-43, IEEE Computer Society, EEUU; 2009.

[Balikuddembe, 2008] Balikuddembe, J.K., Potgieter A.E., Using actor object operations structures to understand project requirements complexities, Proceedings ICSEA'08, pp. 290-294, IEEE Computer Society, EEUU, 2008.

[Banker, 1994] Banker, R.D., Chang, H., Kemerer, C., Evidence on economies of scale in software development, *Information and Software Technology*, Vol. 36, No. 5, pp. 275-282, Elsevier, 1994.

[Basili, 1994] Basili, V. R., Caldiera, G. and Rombach, H. D.: Goal Question Metric Paradigm. *Encyclopedia of Software Engineering*, ed. J. J. Marciniak, Wiley, 1994.

[Beg, 2008] Beg R., Abbas Q., Verma R.P., An approach for requirement prioritization using B-tree, *Proceedings ICETET'08*, pp. 1216-1221, IEEE Computer Society, EEUU, 2008.

[Benyhaia, 1990] Benyhaia, H., Deshamais, J., Hudon, G., Martin, C., Adjustment model for Function Points scope factors-a statistical study. *Ifpug Spring Conference*, Florida, Abril 1990.

[Berander, 2006] Berander P., Jonsson P. Hierarchical Cumulative Voting (HCV) prioritization of requirements in hierarchies, *International Journal of Software Engineering and Knowledge Engineering*, Vol. 16, No. 6, pp.819-849, World Scientific, 2006.

[Biesheuvel, 2008] Biesheuvel, C.J., Vergouwe, Y., Steyerberg, E.W., Grobbee, D.E., Moons, K.G.M.: Polytomous Logistic Regression Analysis Could Be Applied More Often in Diagnostic Research, *Journal of Clinical Epidemiology*, Vol. 61, No. 2, pp. 125-134, Elsevier, 2008.

[Biolchini, 2005] Biolchini, J., Mian, P.G., Natali, A.C.C., and Travassos, G.H., Systematic Review in Software Engineering, *Univ. Rio de Janeiro, TR, ES 679/05*, 2005.

[Bittner, 2003] Bittner, K., *Use Case Modeling*, Addison-Wesley, EEUU, 2003.

[Boehm, 1981] Boehm, B., *Software Engineering Economics*, Prentice Hall, EEUU, 1981.

[Boehm, 2000a] Boehm, B., et al. *Software cost estimation with Cocomo II*, Prentice-Hall, EEUU, 2000.

[Boehm, 2000b], Boehm, B., et al., *Cocomo™ II.2000.0 Model Manual*, disponible en http://csse.usc.edu/csse/research/CocomoII/cocomo_downloads.htm, consultado en Noviembre de 2011.

[Borooah, 2001] Borooah, V.,K., *Logit and Probit: Ordered and Multinomial Models*, Sage Publications, EEUU, 2001.

[Brereton, 2007] Brereton, P., Kitchenham, B., Budgen, D., Turner, D., Khalil, M., Lessons from applying the systematic literature review process within the software engineering domain, *Journal of Systems and Software*, Vol. 80, No. 4, pp. 571-583, Elsevier, 2007.

[Bruegge, 2002] Bruegge B., Allen, S., Ingeniería de Software Orientado a Objetos, Addison-Wesley, EEUU, 2002.

[Cantone, 2004] Cantone G., Pace D., Calavaro G., Applying function point to unified modeling language: Conversion model and pilot study, Proceedings - International Software Metrics Symposium, pp.280-291, IEEE Computer Society, EEUU, 2004.

[Carlshamre, 2001] Carlshamre P., Sandahl, K., Lindvall, M., Regnell, B., Natt och Dag, J., An industrial survey of requirements interdependencies in software product release planning, Proceedings International Symposium on Empirical Software Engineering, pp. 84-92, IEEE Computer Society, EEUU, 2001.

[Carlshamre, 2002] Carlshamre P., Release planning in market-driven software product development: provoking an understanding, Requirements Engineering Vol. 7 No.3 pp.139-151, Springer, Alemania, 2002.

[Carver, 2003] Carver, J., Jaccheri, L., Morasca, S., Issues in Using Students in Empirical Studies in Software Engineering Education, Proceedings METRICS'03, pp. 239-249, IEEE Computer Society, EEUU, 2003.

[Cepeda, 2008] Cepeda, M.S., Chapman, C.R., Miranda, N., Sanchez, R., Rodriguez, C.H., Restrepo, A.E., Ferrer, L.M., Linares, Carr, D.B.: Emotional Disclosure Through Patient Narrative May Improve Pain and Well-Being: Results of a Randomized Controlled Trial in Patients with Cancer Pain, Journal of Pain and Symptom Management, Vol. 35, No. 6, pp 623-631, Elsevier, 2008.

[Chen, 1975] Chen, P., The Entity-Relationship Model: Toward a Unified View of Data, ACM Transactions on Database Systems, Vol. 1, No.1, pp. 9-36, ACM, EEUU, 1976.

[Chen, 2004] Cheng, Ch., Hughes, J.: Using Ordinal Regression Model to Analyze Student Satisfaction Questionnaires, IR Applications, Vol. 1, Association for Institutional Research, 2004.

[Chamundeswari, 2008] Chamundeswari A., Babu C., Function point size estimation for object oriented software based on use case model, ICSOFT 2008 - Proceedings of the 3rd International Conference on Software and Data Technologies, pp. 114-120, INSTICC Press, 2008.

[Coad, 1997] Coad, P., North,D., Mayfield,M., Object Models: Strategies, Patterns and Applications, Second Edition, Prentice-Hall, EEUU, 1997.

[Cockburn, 2000] Cockburn, A., Writing effective use cases, Addison-Wesley, EEUU, 2000.

[Cockburn, 2008] Cockburn, A., Using Both Incremental and Iterative Development, Crosstalk The Journal of Defense Software Engineering, Vol. 21 No.5, <http://www.crosstalkonline.org/>, Mayo 2008.

[Cohn, 2010] Cohn, M., *Succeeding with Agile: Software Development Using Scrum*, Addison-Wesley, EEUU, 2010.

[Condori, 2006] Condori, N., *Un Procedimiento de Medición de Tamaño Funcional para Especificaciones de Requisitos*, Tesis Doctoral, Universidad Politécnica de Valencia, 2006.

[Conte, 1986] Conte, S.D., Dunsmore, H.E., Shen, V.Y., *Software engineering metrics and models*, Benjamin-Cummings, Menlo Park, EEUU, 1986.

[Cox, 1989] Cox, D. R., Snell, E. J., *The Analysis of Binary Data*, 2nd ed. London: Chapman and Hall, 1989.

[Curtis, 1994] Curtis, B., Carleton, A., *Seven \pm Two Software Measurement Conundrum*, Proceedings Second International Software Metrics Symposium, IEEE Computer Society, EEUU, 1993.

[Davey, 2007] Davey, H.M., Barratt, A.L., Butow, P.N., Deeks, J.J., *A rone-item question with a Likert or Visual Analog Scale adequately measured current anxiety*, *Journal of Clinical Epidemiology*, Vol. 60, pp. 356–360, Elsevier, 2007.

[Davis, 1989] Davis, F.D., *Perceived Usefulness, Perceived Ease of Use and User Acceptance of Information Technology*, *MIS Quarterly*, Vol. 13, No. 3, pp. 319-340, 1989.

[Del Bianco, 2008] Del Bianco, V., Lavazza, L. Garavaglia, C., *Using Function Point in the Estimation of Real-Time Software: an Experience*. Software, Measurement European Forum, Milan Italia, May 2008.

[Dyba, 2007] Dyba, T., Dingsoyr, T., Hanssen, G.K., *Applying Systematic Reviews to Diverse Study Types: An Experience Report*, Proceedings ESEM 2007, pp. 225-234, IEEE Computer Society, EEUU, 2007.

[Denne, 2004] Denne, M., Cleland-Huang, J.: *The Incremental Funding Method: Data-Driven Software Development*, *IEEE Software* Vol 21, No. 1, IEEE Computer Society, EEUU, 2004.

[Fetcke, 1997] Fetcke T., Abran A., Nguyen T., *Mapping the OO-Jacobson approach into function point analysis*, Proceedings of the Conference on Technology of Object-Oriented Languages and Systems, TOOLS, pp. 192-202, IEEE Computer Society, EEUU, 1997.

[Engström, 2008] Engström, E., Skoglund, M., Runeson, P., *Empirical evaluations of regression test selection techniques: a systematic review*, Proceedings ESEM 2008, ACM, EEUU, 2008.

[Eriksson, 2004] Eriksson, H., Penker, M., Lyons, B., Fado, D., *UML 2 Toolkit*, Wiley Publishing, EEUU, 2004.

[Fenton, 1997] N.E. Fenton, F.E., Pfleeger, S.L., Software Metrics: A Rigorous and Practical Approach, second edition, PWS, Inglaterra, 1997.

[Fienberg, 1999] Fienberg, S., Albert, J. H., Lievesley, D., Ordinal Data Modeling, Springer-Verlag, EEUU, 1999.

[Fowler, 1997] Fowler, M., Analysis patterns – reusable object models, Addison Wesley, EEUU, 1997.

[Gaur, 2010] Gaur V., Soni A., Bedi P., An application of multi-person decision-making model for negotiating and prioritizing requirements in agent-oriented paradigm, Proceedings DSDD 2010, pp. 164-168, IEEE Computer Society, EEUU, 2010.

[Gomez, 2010] Gomez, A., Rueda, G., Alarcón, P.P., A Systematic and Lightweight Method to Identify Dependencies between User Stories, Agile Processes in Software Engineering and Extreme Programming, Lecture Notes in Business Information Processing, Volume 48, Part 2, pp. 190-195, Springer, Alemania, 2010.

[Gruenbacher, 2000] Gruenbacher, P., Collaborative requirements negotiation with EasyWinWin, Proceedings 11th International Workshop on Database and Expert Systems Applications, pp. 954-958, IEEE Computer Society, EEUU, 2000.

[Harput, 2005] Harput V., Kaindl H., Kramer S., Extending function point analysis to object-oriented requirements specifications, Proceedings - International Software Metrics Symposium, pp. 39-47, IEEE Computer Society, EEUU, 2005.

[Hastings, 2001] Hastings, T.E., Sajeev, A.S.M., A Vector-Based Approach to Software Size Measurement and Effort Estimation, IEEE Transactions on Software Engineering, Vol. 27, No. 4, pp. 337-349, IEEE Computer Society, EEUU, 2001.

[Hericko, 2007] Hericko M., Zivkovic A., The size and effort estimates in iterative development, Information and Software Technology, Vol. 80, No.7-8, pp. 772-781, Elsevier, 2007.

[IEEE, 1998] IEEE Computer Society, IEEE Std. 830-1998: Recommended Practice for Software Requirements Specifications, The Institute of Electrical and Electronics Engineers, EEUU, 1998.

[Ifpug, 1999] Ifpug, Function Points Counting Practices Manual (version 4.1.1), Ifpug: International Function Point User Group, Westerville Ohio, 1999.

[Ifpug, 2004] Ifpug, Function Points Counting Practices Manual (version 4.2.1), Ifpug: International Function Point User Group, Westerville Ohio, 2004.

[Ifpug, 2009] Ifpug, Function Points Counting Practices Manual (version 4.3), Ifpug: International Function Point User Group, Westerville Ohio, 2004.

[IBM, 2006] IBM, Rational Unified Process version 2001A.04.00.13, EEUU, 2006.

[ISO, 1998] ISO. ISO/IEC TR 15271 - Guide for ISO/IEC 12207 (Software Life Cycle Processes), 1998.

[ISO, 2003] ISO. ISO/IEC 14143-3 - Information technology - Software measurement - Functional size measurement - Part 3: Verification of functional size measurement methods, 2003.

[Jacobson, 1992] Jacobson, I.: Object-Oriented Software Engineering. A Use Case Driven Approach, Addison-Wesley, EEUU, 1992.

[Jaaksi, 1998] Jaaksi, A., A Method for Your Object-Oriented Project, Journal of Object-Oriented Programming, Vol 10. No 9, 1998.

[Jaaksi, 1999] Jaaksi, A., Aalto, J., Aalto, A., Vatto, K., Tried & True Object Development: Industry-Proven Approaches with UML, Cambridge University Press, Inglaterra, 1999.

[Jeffery, 1993] Jeffery, D.R., Low, G.C., Barnes, M., A Comparison of Function Point Counting Techniques, IEEE Transactions on Software Engineering, Vol. 19, Nº. 5, pp. 529-532, IEEE, EEUU, 1993.

[Jorgensen, 2007] Jorgensen, M., Shepperd, M., A Systematic Review of Software Development Cost Estimation Studies, IEEE Transactions on Software Engineering, Vol. 33, No. 1, pp. 33 -53, IEEE, EEUU, 2007.

[Juristo, 2001] Juristo, N., Moreno, A.M.: Basics of Software Engineering Experimentation. Kluwer Academic Publishers, Boston, 2001.

[Kang, 2010] Kang, S., Choi, O., Baik, J., Model-Based Dynamic Cost Estimation and Tracking Method for Agile Software Development, Proceedings IEEE/ACIS 9th International Conference on Computer and Information Science, pp.743-748, IEEE Computer Society, EEUU, 2010.

[Kalman, 1960] Kalman, R.E., A new approach to linear filtering and prediction problems, Transactions of the ASME-Journal of Basic Engineering, Vol. 82 Nº 1, pp. 35-45, ASME, EEUU, 1960.

[Karlsson, 2007] Karlsson L., Thelin T., Regnell B., Berander P., Wohlin C., Pair-wise comparisons versus planning game partitioning-experiments on requirements prioritisation techniques, Empirical Software Engineering, Vol. 12, No. 1, pp. 3-33, Springer, Alemania, 2007.

[Kendall, 2005] Kendall, K., Kendall, J., Systems Analysis and Design, Sixth Edition, Prentice Hall, USA, 2005.

[Kitchenham, 2004] Kitchenham, B., Procedures for Performing Systematic Reviews, Keele University, Technical Report TR/SE-0401 and NICTA Technical Report 0400011T.1.

[Kitchenham, 2007] Kitchenham, B., Guidelines for Performing Systematic Literature Review in Software Engineering. EBSE Technical Report, Version 2.3, Keele University, 2007.

[Kodanagallur, 2006] Kodanagallur, V., Shim, S., Analysis patterns: a taxonomy and its implications, Information Systems Management, Vol. 23, Nº3, pp. 52-61, Taylor & Francis, 2006.

[Koh, 2008] Koh, T.W., Selamat, M.H., Ghani, A.A.A., Exponential Effort Estimation Model Using Unadjusted Function Points, Information Technology Journal, Vol. 7 No. 6, pp. 830-839, Asian Network for Scientific Information, 2008.

[Kundu, 2007] Kundu, D., Samanta, D.: A Novel Approach of Prioritizing Use Case Scenarios. In APSEC 2007, pp. 542-549, IEEE Computer Society, EEUU (2007).

[Larman, 1999] Larman, C., UML y Patrones, Introducción al Análisis y Diseño Orientado a Objetos, México, Prentice Hall Hispanoamericana, 1999

[Lavazza, 2008] Lavazza L.A., Bianco V.D., Garavaglia C., Model-based functional size measurement, ESEM'08: Proceedings of the 2008 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, pp. 100-109, ACM, EEUU, 2008.

[Leek, 1997] Leek, C., Information systems frameworks and strategy, Industrial Management & Data Systems, Vol. 97, No. 3, pp.86-89, doi:10.1108/02635579710173149, 1997.

[Leffingwell, 2011] Leffingwell, D., Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise, Pearson Education, EEUU, 2011.

[Likert, 1931] Likert, R. A technique for the measurement of attitudes. Archives of Psychology. New York, Columbia University Press, 1931.

[Lokan, 1999] Lokan, C., Abran, A., Multiple viewpoints in functional size measurement, International Workshop on Software Measurement (IWSM'99), pp. 122-132, Lac Superieur, Canada, 1999.

[Lokan, 2000] Lokan, C.J., An empirical analysis of function point adjustment factors, Information and Software Technology, Vol. 42, No. 9, pp. 649-659, Elsevier, 2000.

[Long, 1997] Long, S., Regression Models for Categorical and Limited Dependent Variables, SAGE Publications, EEUU, 1997.

[Longstreet, 2001] Longstreet, D., Use Case and Function Points, <http://www.softwaremetrics.com/>, Longstreet Consulting Inc, USA, 2001.

[Lucas, 2009] Lucas, F., Molina, F., Toval, J., A systematic review of UML model consistency management. *Information & Software Technology*, Vol. 51, No. 12, pp. 1631-1645, Elsevier, 2009.

[MAP, 2000] Ministerio de Administraciones Públicas, Análisis del Sistema de Información-Métrica Versión 3, España, 2000.

[MacDonnell, 2007] MacDonell, S., Shepperd, S. Comparing Local and Global Software Effort Estimation Models - Reflections on a Systematic Review. *Proceedings ESEM 2007*, pp.401-409, IEEE Computer Society, EEUU, 2007.

[McCullagh, 1989] McCullagh, P., Nelder, J.A, *Generalized Linear Models*, 2nd ed., Chapman & Hall, Londres, 1989.

[McFadden, 1974] McFadden, D., Conditional logit analysis of qualitative choice behavior. In: *Frontiers in Economics*, P. Zarembka, Academic Press, EEUU, 1974.

[Mohamed, 2008] Mohamed A.S.I., El-Maddah B.I.A., Wahba C.A.M., Criteria-based requirements prioritization for software product management, *SERP 2008*, pp. 587-593, CSREA Press, 2008.

[Moisiadis, 2001] Moisiadis, F.: Prioritizing Scenario Evolution. In: *ICRE 2000*, pp. 85--94, IEEE Computer Society, EEUU, 2001.

[Moisiadis, 2000] Moisiadis, F.: Prioritising Use Cases and Scenarios. In: *TOOLS'37-2000*, pp. 108--119, IEEE Computer Society, EEUU, 2000.

[Moody, 2001] Moody D. L., *Dealing with Complexity: A Practical Method for Representing Large Entity Relationship Models*, PhD. Thesis, Department of Information Systems, University of Melbourne, Australia, 2001.

[Muijs, 2004] Muijs, D., *Doing Quantitative Research in Education with SPSS*, Sage Publications, EEUU, 2004.

[Mulrow, 1998] Mulrow, C., Cook, D., *Systematic Reviews: Synthesis of Best Evidence for Health Care Decisions*, Philadelphia: American College of Physicians, 1998.

[Nagelkerke, 1991] Nagelkerke, N. J. D., A note on the general definition of the coefficient of determination, *Biometrika*, Vol. 78, No. 3, pp. 691-692, Oxford Journals, Inglaterra, 1991.

[Norris, 2006] Norris, C.M., Ghali, W.A., Duncan Saunders, L., Brant, R., Galbraith, D., Faris, P., Knudtson, M.L.: Ordinal Regression Model and the Linear Regression Model were Superior to the Logistic Regression Models, *Journal of Clinical Epidemiology*, Vol. 59, No. 5, pp. 448-456, Elsevier, 2006.

[Nunally, 1978] Nunally J., Psychometric Theory, McGraw-Hill, 2nd ed., New York, 1978.

[OMG, 2008] Object Management Group, OMG Unified Modeling Language, <http://www.uml.org>, EEUU (2008).

[Orr, 2000] Orr, G., Reeves, T.E., Function point counting: one program's experience, Journal of Systems and Software, Vol. 53, No. 3, pp. 239-244, Elsevier, 2000.

[Pastor, 2001] Pastor, O., Gomez, J., Insfran, E., Pelechano, V., The OO-method approach for information systems modelling: from object-oriented conceptual modeling to automated programming, Information Systems, Vol. 26, No. 7, pp. 507-534, Elsevier, 2001.

[Peeters, 2007] Peeters, J.; Dyson, P., Cost-Effective Security, IEEE Security & Privacy, Vol. 5, No. 3, pp. 85 – 87, IEEE, 2007.

[Pender, 2003] Pender, T., UML Bible, John Wiley & Sons, EEUU, 2003.

[Perini, 2007] Perini A., Ricca F., Susi A., Bazzanella C., An empirical study to compare the accuracy of AHP and CBRanking techniques for requirements prioritization, Proceedings CERE '07, pp. 23-25, IEEE Computer Society, 2007.

[Petersen, 2011] Pettersen, K., Measuring and predicting software productivity: A systematic map and review, Information and Software Technology, Vol. 53, No. 4, pp. 317-343, Elsevier, 2011.

[Petticrew, 2006] Petticrew, M., Roberts, H., Systematic Reviews in the Social Sciences: A Practical Guide, Blackwell Publishing, 2006.

[Piattini, 2004] Piattini, M., Calvo-Manzano, J.A, Cervera, J., Fernandez, L., Análisis y Diseño de Aplicaciones Informáticas de Gestión: Una Perspectiva de Ingeniería del Software, Alfaomega, México, 2004.

[Pfleeger, 2002] Pfleeger, S., Ingeniería de Software: Teoría y Práctica, Prentice Hall, Buenos Aires, Argentina, 2002.

[Pow-Sang, 2007] Pow-Sang, J.A, Imbert, R., Including the Composition Relationship among Classes to Improve Function Points Analysis, Proceedings Jornadas Peruanas de Computación 2007, JPC'07, Trujillo, Peru.

[Pow-Sang, 2008] Pow-Sang, J.A., Nakasone, A., Imbert, R., Moreno, A.M., Case Study Evaluations for a Function Point Counting Improvement for Object-Oriented Projects, Proceedings Mexican International Conference on Computer Science 2008, México, 2008.

[PMI, 2008] Project Management Institute, A Guide to the Project Management Body of Knowledge: Pmbok guide – Fourth Edition, PMI, EEUU, 2008.

[Rosenberg, 1999] Rosenberg, D., Scott, K.: Use Case Driven Object Modeling with UML, Addison-Wesley, Massachusetts, EEUU, 1999.

[Royce, 1970] Royce, W. W., Managing the Development of Large Software Systems: Concepts and Techniques. Proceedings WESCON, IEEE, EEUU 1970.

[Ruhe, 2005] Ruhe G., Momoh J., Strategic release planning and evaluation of operational feasibility, EEUU, 2005.

[Rumbaugh, 1991] Rumbaugh, J., Blaha, M., Premerlani, W., Frederick, E., Lorensen, W., Object Oriented Modeling and Design, Prentice Hall, EEUU, 1991.

[Runeson, 1998] Runeson, P., Regnell, B., Derivation of an integrated operational profile and use case model

[Ryaz, 2009a] Riaz, M., Mendes, E., Tempero, E., A Systematic Review of Software Maintainability Prediction and Metrics, Proceedings ESEM 2009, IEEE Computer Society, EEUU, 2009.

[Ryaz, 2009b] Riaz, M., Systematic Review of Software Maintainability Prediction and Metrics, <http://www.cs.auckland.ac.nz/~mehwish/SLR>, 2009.

[Ryser, 2000] Ryser, J., Glinz, M., Using Dependency Charts to Improve Scenario-Based Testing. In: 17th International Conference on Testing Computer Software TCS'2000, Washington D.C., EEUU, 2000.

[Saliu, 2007] Saliu, M.O., Ruhe, G., Bi-objective release planning for evolving software systems, Proceedings of the the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering, pp. 105-114, ACM, EEUU, 2007.

[Sadoski, 1997] Sadoski, D., Two tier software architectures, C4 Software Technology Reference Guide-A Prototype, Handbook CMU/SEI-97-HB-001, 1997, pp. 381-386, Software Engineering Institute, 1997.

[Sapna, 2009] Sapna, P.G.; Mohanty, H., Prioritizing Use Cases to Aid Ordering of Scenarios, Proceedings Third UKSim European Symposium on Computer Modeling and Simulation, EMS '09, pp. 136-141, IEEE, EEUU, 2009.

[Shneider, 2001] Shneider, G., Winters, J., Applying use cases, Second Edition, Addison Wesley, EEUU, 2001.

[SEI, 2010] Software Engineering Institute (SEI), CMMI for Development, Version 1.3, Disponible en <http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm>

[Senn, 1992] Senn, J., Análisis y Diseño de Sistemas de Información, McGraw Hill, Mexico, 1992.

[Silverston, 2001] Silverston, L., The data model resource book, Vol 1, A Library Universal Data Models for All Enterprises, John Wiley, EEUU, 2001.

[Some, 2007] Some, S.: Specifying Use Case Sequencing Constraints using Description Elements, Proceedings Sixth International Workshop on Scenarios and State Machines (SCESM'07), pp. 4, IEEE Computer Society, EEUU, 2007.

[Sommerville, 2011] Sommerville, I., Ingeniería de Software, 9na edición, Pearson Educación, México, 2011.

[Strasunkas, 2006] Strasunkas, D., Domain Model-Centric Distributed Development: An approach to semantics-based change impact management, Tesis Doctoral, Norwegian University of Science and Technology, 2006.

[Szoke, 2008] Szoke, A., A Proposed Method for Release Planning from Use Case-based Requirements Specification, Proceedings 34th Euromicro Conference Software Engineering and Advanced Applications, 2008. SEAA '08, pp. 449 – 456, IEEE, EEUU, 2008.

[Temel, 2006] Temel, J.S., Pirl, W.F., Recklitis, C.J.: Feasibility and validity of a one-item fatigue screen in a thoracic oncology clinic, Journal of Thoracic Oncology, Vol. 1, No. 5, Lippincott Williams & Wilkins, 2006.

[Uemura, 2001] Uemura T., Kusumoto S., Inoue K., Function-point analysis using design specifications based on the unified modelling language, Journal of Software Maintenance and Evolution: Research and Practice, Vol. 13, No. 4, pp.223–243, Wiley, 2001.

[Van Den Berg, 1999] Van Den Berg, K.G., Simons, A.J.H., Control-flow semantics of use cases in UML, Information and Software Technology, Vol. 41, No. 10, pp. 651-659, Elsevier, 1999.

[Webster, 2002] Webster, J., Watson, R. Analyzing the past to prepare for the future: Writing a literature review, MIS Quarterly Vol. 26 No. 2, pp. xiii-xxiii, EEUU, 2002.

[Whittle, 2006] Whittle, J., Specifying Precise Use Cases with Use Case Charts, Proceedings Satellite Events at the MoDELS 2005 Conference, Lecture Notes in Computer Science, 2006, Volume 3844/2006, pp. 290-301, Springer, Alemania, 2006.

[Wohed, 2001] Wohed, P., Conceptual patterns- a consolidation of Coad's and Wohed's approaches, Natural Language Processing and Information Systems, Lecture Notes in Computer Science, Volume 1959/2001, pp. 340-351, Springer, Alemania, 2001.

[Yourdon, 1989] Yourdon, E., Modern Structured Analysis, Yourdon Press, EEUU, 1989.

[Zhang, 2005] Zhang, W., Mei, H., Zhao, H., A feature-oriented approach to modeling requirements dependencies, Proceedings. 13th IEEE International Conference on Requirements Engineering, doi: 10.1109/RE.2005.6, pp. 273 – 282, IEEE, EEUU; 2005.

[Zhang, 2007] Zhang W., Han, D., Kunz, T., Hansen, K.M., Mobile game development: object orientation or not, Proceedings of the International Computer Software and Applications Conference (COMPSAC'07), pp. 601-608, IEEE Computer Society, EEUU, 2007.

[Zivkovic, 2005] Zivkovic A., Rozman I., Hericko M., Automated software size estimation based on function points using UML models, Information and Software Technology, Vol. 47, No. 13, pp. 881-890, Elsevier, 2005.

Información Complementaria del Capítulo 2

Anexo

A

En este anexo se incluye información adicional relacionada al capítulo 2 correspondiente al estado de la cuestión. Las secciones A.1, A.2, A.3 y A.4 muestran las cadenas de búsquedas empleadas en cada base de datos. La Sección A.5 muestra la relación de publicaciones que se obtuvieron para la pregunta de investigación p2c.

Cabe resaltar que la interfaz de Computer Database no permite el ingreso de cadenas completas de búsqueda. En dicha interfaz se tuvieron que ingresar los operadores lógicos como **OR** o **AND** mediante listas desplegables. También, se tuvo que indicar, mediante listas desplegables, si la búsqueda se va realizar por *abstract*, *title* o *keyword*. Por ello, las cadenas que se incluyen en el capítulo 2, se tuvieron que cortar para que puedan ser ingresadas en la interfaz de Computer Database. En el caso de las otras bases de datos, las cadenas se tuvieron que adaptar a la sintaxis que especifica cada una de ellas, porque permitían el ingreso de cadenas completas de búsqueda. Debido a ello, solo se incluyen en este anexo las cadenas de búsquedas de todas las bases de datos excepto Computer Database.

A.1 Cadenas de Búsqueda para la Pregunta 2

Las cadenas de búsqueda para la pregunta de investigación 2 ¿qué técnicas/métodos/aproximaciones basadas en Puntos de función han sido utilizadas para estimar y planificar desarrollos de software orientado a objetos que empleen ciclos de vida incremental? son las siguientes:

- Scopus:

```
TITLE-ABS-KEY(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*")) AND ("object orient*" OR "UML" OR "unified modeling language"))
```

- ISI Web of Science:

```
TS=(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language") AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*"))
```

TI=(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language") AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*"))

- Science Direct:

TITLE-ABS-KEY(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*") AND ("object orient*" OR "UML" OR "unified modeling language"))

- IEEE Xplore

((("Document Title": "software" OR "Document Title": "application" OR "Document Title": "applications" OR "Document Title": "system" OR "Document Title": "systems") AND ("Document Title": "development" OR "Document Title": "construction" OR "Document Title": "project" OR "Document Title": "projects" OR "Document Title": "process" OR "Document Title": "processes" OR "Document Title": "engineering")) AND ("Document Title": "object oriented" OR "Document Title": "object orientation" OR "Document Title": "UML" OR "Document Title": "Unified Modeling Language") AND ("Document Title": "function point" OR "Document Title": "function points" OR "Document Title": "Ifpug" OR "Document Title": "NESMA") AND ("Document Title": "increment" OR "Document Title": "increments" OR "Document Title": "incremental" OR "Document Title": "iteration" OR "Document Title": "iterations" OR "Document Title": "iterative" OR "Document Title": "sprint" OR "Document Title": "sprints" OR "Document Title": "release" OR "Document Title": "releases") AND ("Document Title": "estimat*" OR "Document Title": "forecast*" OR "Document Title": "calculat*" OR "Document Title": "predict*" OR "Document Title": "plan*"))

((("Abstract": "software" OR "Abstract": "application" OR "Abstract": "applications" OR "Abstract": "system" OR "Abstract": "systems") AND ("Abstract": "development" OR "Abstract": "construction" OR "Abstract": "project" OR "Abstract": "projects" OR "Abstract": "process" OR "Abstract": "processes" OR "Abstract": "engineering")) AND ("Abstract": "object oriented" OR "Abstract": "object orientation" OR "Abstract": "UML" OR "Abstract": "Unified Modeling Language") AND ("Abstract": "function point" OR "Abstract": "function points" OR "Abstract": "Ifpug" OR "Abstract": "NESMA") AND ("Abstract": "increment" OR "Abstract": "increments" OR "Abstract": "incremental" OR "Abstract": "iteration" OR "Abstract": "iterations" OR "Abstract": "iterative" OR "Abstract": "sprint" OR "Abstract": "sprints" OR "Abstract": "release" OR "Abstract": "releases") AND ("Abstract": "estimat*" OR "Abstract": "forecast*" OR "Abstract": "calculat*" OR "Abstract": "predict*" OR "Abstract": "plan*"))

- ACM:

((("Abstract": "function point" OR "Abstract": "function points" OR "Abstract": "Ifpug" OR "Abstract": "NESMA") AND ("Abstract": "increment" OR "Abstract": "increments" OR "Abstract": "incremental" OR "Abstract": "iteration" OR "Abstract": "iterations" OR "Abstract": "iterative" OR "Abstract": "sprint" OR "Abstract": "sprints" OR "Abstract": "release" OR "Abstract": "releases") AND ("Abstract": "estimat*" OR "Abstract": "forecast*" OR "Abstract": "calculat*" OR "Abstract": "predict*" OR "Abstract": "plan*") AND ("Abstract": "object orient*" OR "Abstract": "UML" OR "Abstract": "unified modeling language"))

- Engineering Village

((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language") AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND (estimat* OR forecast* OR calculat* OR predict* OR plan*)) WN KY

A.2 Cadenas de Búsqueda para la Pregunta 2a

Para la pregunta de investigación 2a ¿qué técnicas o modelos de conversión se han definido para calcular puntos de función empleando modelos orientados a objetos?, las cadenas de búsqueda empleadas por base de datos fueron las siguientes:

- Scopus:

TITLE-ABS-KEY(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language") AND ("function point" OR "function points" OR "Ifpug" OR "NESMA"))

- ISI Web of Science:

TS=(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language")AND ("function point" OR "function points" OR "Ifpug" OR "NESMA"))

TI=(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language")AND ("function point" OR "function points" OR "Ifpug" OR "NESMA"))

- Science Direct:

TITLE-ABS-KEY(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language") AND ("function point" OR "function points" OR "Ifpug" OR "NESMA"))

- IEEE Xplore:

((("Abstract": "software" OR "Abstract": "application" OR "Abstract": "applications" OR "Abstract": "system" OR "Abstract": "systems") AND ("Abstract": "development" OR "Abstract": "construction" OR "Abstract": "project" OR "Abstract": "projects" OR "Abstract": "process" OR "Abstract": "processes" OR "Abstract": "engineering")) AND ("Abstract": "object oriented" OR "Abstract": "object orientation" OR "Abstract": "UML" OR "Abstract": "Unified Modeling Language")AND ("Abstract": "function point" OR "Abstract": "function points" OR "Abstract": "Ifpug" OR "Abstract": "NESMA"))

((("Document Title": "software" OR "Document Title": "application" OR "Document Title": "applications" OR "Document Title": "system" OR "Document Title": "systems") AND ("Document Title": "development" OR "Document Title": "construction" OR "Document Title": "project" OR "Document Title": "projects" OR "Document Title": "process" OR "Document Title": "processes" OR "Document Title": "engineering")) AND ("Document Title": "object oriented" OR "Document Title": "object orientation" OR "Document Title": "UML" OR "Document Title": "Unified Modeling Language")AND ("Document Title": "function point" OR "Document Title": "function points" OR "Document Title": "Ifpug" OR "Document Title": "NESMA"))

- ACM:

((("Abstract": "software" OR "Abstract": "application" OR "Abstract": "applications" OR "Abstract": "system" OR "Abstract": "systems") AND ("Abstract": "development" OR "Abstract": "construction" OR "Abstract": "project" OR "Abstract": "projects" OR "Abstract": "process" OR "Abstract": "processes" OR "Abstract": "engineering")) AND ("Abstract": "object oriented" OR "Abstract": "object orientation" OR "Abstract": "UML" OR "Abstract": "Unified Modeling Language")AND ("Abstract": "function point" OR "Abstract": "function points" OR "Abstract": "Ifpug" OR "Abstract": "NESMA"))

- Engineering Village:

((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("object oriented" OR "object orientation" OR "UML" OR "Unified Modeling Language")) AND ("function point" OR "function points" OR "Ifpug" OR "NESMA")) WN KY

A.3 Cadenas de Búsqueda para la Pregunta 2b

Para la pregunta de investigación 2b ¿qué actividades/tareas/pasos se han definido para calcular el esfuerzo requerido para cada incremento?, las cadenas de búsqueda empleadas por base de datos fueron las siguientes:

- Scopus:

TITLE-ABS-KEY(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*"))

- ISI Web of Science:

TS=(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*"))

TI=(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*"))

- Science Direct:

TITLE-ABS-KEY(((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) AND ("function point" OR "function points" OR "Ifpug" OR "NESMA") AND ("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") AND ("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*"))

- IEEE Xplore:

((("Abstract": "software" OR "Abstract": "application" OR "Abstract": "applications" OR "Abstract": "system" OR "Abstract": "systems") AND ("Abstract": "development" OR "Abstract": "construction" OR "Abstract": "project" OR "Abstract": "projects" OR "Abstract": "process" OR "Abstract": "processes" OR "Abstract": "engineering")) AND ("Abstract": "function point" OR "Abstract": "function points" OR "Abstract": "Ifpug" OR "Abstract": "NESMA") AND ("Abstract": "increment" OR "Abstract": "increments" OR "Abstract": "incremental" OR "Abstract": "iteration" OR "Abstract": "iterations" OR "Abstract": "iterative" OR "Abstract": "sprint" OR "Abstract": "sprints" OR "Abstract": "release" OR "Abstract": "releases") AND ("Abstract": "estimat*" OR "Abstract": "forecast*" OR "Abstract": "calculat*" OR "Abstract": "predict*" OR "Abstract": "plan*"))

((("Document Title":"software" OR "Document Title":"application" OR "Document Title":"applications" OR "Document Title":"system" OR "Document Title":"systems") AND ("Document Title":"development" OR "Document Title":"construction" OR "Document Title":"project" OR "Document Title":"projects" OR "Document Title":"process" OR "Document Title":"processes" OR "Document Title":"engineering")) AND ("Document Title":"function point" OR "Document Title":"function points" OR "Document Title":"Ifpug" OR "Document Title":"NESMA") AND ("Document Title":"increment" OR "Document Title":"increments" OR "Document Title":"incremental" OR "Document Title":"iteration" OR "Document Title":"iterations" OR "Document Title":"iterative" OR "Document Title":"sprint" OR "Document Title":"sprints" OR "Document Title":"release" OR "Document Title":"releases") AND ("Document Title":"estimat*" OR "Document Title":"forecast*" OR "Document Title":"calculat*" OR "Document Title":"predict*" OR "Document Title":"plan*"))

- ACM:

((("Abstract":"software" OR "Abstract":"application" OR "Abstract":"applications" OR "Abstract":"system" OR "Abstract":"systems") AND ("Abstract":"development" OR "Abstract":"construction" OR "Abstract":"project" OR "Abstract":"projects" OR "Abstract":"process" OR "Abstract":"processes" OR "Abstract":"engineering")) AND ("Abstract":"function point" OR "Abstract":"function points" OR "Abstract":"Ifpug" OR "Abstract":"NESMA") AND ("Abstract":"increment" OR "Abstract":"increments" OR "Abstract":"incremental" OR "Abstract":"iteration" OR "Abstract":"iterations" OR "Abstract":"iterative" OR "Abstract":"sprint" OR "Abstract":"sprints" OR "Abstract":"release" OR "Abstract":"releases") AND ("Abstract":"estimat*" OR "Abstract":"forecast*" OR "Abstract":"calculat*" OR "Abstract":"predict*" OR "Abstract":"plan*"))

- Engineering Village:

((("software" OR "application" OR "applications" OR "system" OR "systems") AND ("development" OR "construction" OR "project" OR "projects" OR "process" OR "processes" OR "engineering")) WN KY) AND (("function point" OR "function points" OR "NESMA") WN KY) AND (("increment" OR "increments" OR "incremental" OR "iteration" OR "iterations" OR "iterative" OR "sprint" OR "sprints" OR "release" OR "releases") WN KY) AND (("estimat*" OR "forecast*" OR "calculat*" OR "predict*" OR "plan*") WN KY))

A.4 Cadenas de Búsqueda para la Pregunta 2c

Para la pregunta de investigación 2c ¿qué actividades/tareas/pasos se han definido para determinara la prioridad en la construcción de casos de uso?, las cadenas de búsqueda empleadas por base de datos fueron las siguientes:

- Scopus:

TITLE-ABS-KEY(("increment" OR "increments" OR "sprint" OR "sprints" OR "release" OR "releases" OR "user story" OR "user stories" OR "use case" OR "use cases") AND ("priorit*" OR "precede*") AND ("software development" OR "software construction" OR "software project" OR "software projects" OR "software process" OR "software processes" OR "software engineering" OR "requirement" engineering"))

- ISI Web of Science:

TS=(("increment" OR "increments" OR "sprint" OR "sprints" OR "release" OR "releases" OR "user story" OR "user stories" OR "use case" OR "use cases") AND ("priorit*" OR "precede*") AND ("software development" OR "software construction" OR "software project" OR "software projects" OR "software process" OR "software processes" OR "software engineering" OR "requirement" engineering"))

TI=(("increment" OR "increments" OR "sprint" OR "sprints" OR "release" OR "releases" OR "user story" OR "user stories" OR "use case" OR "use cases") AND ("priorit*" OR "precede*") AND ("software development" OR "software construction" OR "software project" OR "software projects" OR "software process" OR "software processes" OR "software engineering" OR "requirement" engineering"))

- Science Direct:

TITLE-ABS-KEY(("increment" OR "increments" OR "sprint" OR "sprints" OR "release" OR "releases" OR "user story" OR "user stories" OR "use case" OR "use cases") AND ("priorit*" OR "precede*") AND ("software development" OR "software construction" OR "software project" OR "software projects" OR "software process" OR "software processes" OR "software engineering" OR "requirement* engineering"))

- IEEE Xplore:

((("Abstract": "increment" OR "Abstract": "increments" OR "Abstract": "sprint" OR "Abstract": "sprints" OR "Abstract": "release" OR "Abstract": "releases" OR "Abstract": "user story" OR "Abstract": "user stories" OR "Abstract": "use case" OR "Abstract": "use cases") AND ("Abstract": "priorit*" OR "Abstract": "precede*") AND ("Abstract": "software development" OR "Abstract": "software construction" OR "Abstract": "software project" OR "Abstract": "software projects" OR "Abstract": "software process" OR "Abstract": "software processes" OR "Abstract": "software engineering" OR "Abstract": "requirement* engineering"))

((("Document Title": "increment" OR "Document Title": "increments" OR "Document Title": "sprint" OR "Document Title": "sprints" OR "Document Title": "release" OR "Document Title": "releases" OR "Document Title": "user story" OR "Document Title": "user stories" OR "Document Title": "use case" OR "Document Title": "use cases") AND ("Document Title": "priorit*" OR "Document Title": "precede*") AND ("Document Title": "software development" OR "Document Title": "software construction" OR "Document Title": "software project" OR "Document Title": "software projects" OR "Document Title": "software process" OR "Document Title": "software processes" OR "Document Title": "software engineering" OR "Document Title": "requirement* engineering"))

- ACM:

((("Abstract": "increment" OR "Abstract": "increments" OR "Abstract": "sprint" OR "Abstract": "sprints" OR "Abstract": "release" OR "Abstract": "releases" OR "Abstract": "user story" OR "Abstract": "user stories" OR "Abstract": "use case" OR "Abstract": "use cases") AND ("Abstract": "priorit*" OR "Abstract": "precede*") AND ("Abstract": "software development" OR "Abstract": "software construction" OR "Abstract": "software project" OR "Abstract": "software projects" OR "Abstract": "software process" OR "Abstract": "software processes" OR "Abstract": "software engineering" OR "Abstract": "requirement* engineering"))

- Engineering Village:

((("increment" OR "increments" OR "sprint" OR "sprints" OR "release" OR "releases" OR "user story" OR "user stories" OR "use case" OR "use cases") AND (priorit* OR precede*) AND ("software development" OR "software construction" OR "software project" OR "software projects" OR "software process" OR "software processes" OR "software engineering" OR "requirement engineering" OR "requirements engineering")) WN KY

A.5 Resultados obtenidos para la búsqueda de la pregunta 2c

La relación de publicaciones que se obtuvieron en la búsqueda relacionada a la pregunta 2c se detalla en la siguiente tabla.

Tabla A-1. Publicaciones obtenidas para la pregunta 2c

Autores	Título	Tipo de publicación	Año
Van Valkenhoef G., Tervonen T., De Brock B., Postmus D.	Quantitative release planning in extreme programming	Journal	2011
Del Sagrado J., Aguila I.M., Orellana F.J.	Requirements interaction in the next release problem	Conferencia	2011
Benestad H.C., Hannay J.E.	A comparison of model-based and judgment-based release planning in incremental software projects	Conferencia	2011
Czerwonka J., Das R., Nagappan N., Tarvo A., Teterev A.	CRANE: Failure prediction, change analysis and test prioritization in practice - Experiences from windows	Conferencia	2011
Shihab, E; Jiang, ZM; Adams, B; Hassan, AE; Bowerman, R	Prioritizing the creation of unit tests in legacy software systems	Journal	2011
Zhang, YY; Harman, M; Finkelstein, A; Mansouri, SA	Comparing the performance of metaheuristics for the analysis of multi-stakeholder tradeoffs in requirements optimisation	Journal	2011
Aasem M., Ramzan M., Jaffar A.	Analysis and optimization of software requirements prioritization techniques	Conferencia	2010
Danesh A.S., Ahmad R.	Evaluation of release planning improvement activity in PARSEQ method	Conferencia	2010
Ernst N.A., Mylopoulos J., Borgida A., Jureta I.J.	Reasoning with optional and preferred requirements	Conferencia	2010
Stock D., Wortmann F., Mayer J.H.	Use cases for business metadata - A viewpoint-based approach to structuring and prioritizing business needs	Conferencia	2010
Tonella P., Susi A., Palma F.	Using interactive GA for requirements prioritization	Conferencia	2010
Marjaie S.A., Kulkarni V.	Recognition of hidden factors in requirements prioritization using factor analysis	Conferencia	2010
Christensen H.B., Hansen K.M., Lindstrom B.	Lightweight and continuous architectural software quality assurance using the aSQA technique	Conferencia	2010
Ranganathan P., Magel K.	Understanding requirement engineering (REQ) from a software agent modeling perspective	Conferencia	2010
Ahmad A., Goransson M., Shahzad A.	Limitations of the analytic hierarchy process technique with respect to geographically distributed stakeholders	Conferencia	2010
Gaur V., Soni A., Bedi P.	An application of multi-person decision-making model for negotiating and prioritizing requirements in agent-oriented paradigm	Conferencia	2010
Gaur V., Soni A., Bedi P.	An agent-oriented approach to requirements engineering	Conferencia	2010
Woodward E.V., Bowers R., Thio V.S., Johnson K., Srihari M., Bracht C.J.	Agile methods for software practice transformation	Journal	2010
Helming J., Koegel M., Hodaie Z.	Towards automation of iteration planning	Conferencia	2009
Vahaniitty J., Lassenius C., Rautiainen K., Pekkanen P.	Long-term planning of development efforts by roadmapping - A model and experiences from small software companies	Conferencia	2009
Tourwe T., Codenie W., Boucart N., Blagojevic V.	Demystifying release definition: From requirements prioritization to collaborative value quantification	Conferencia	2009
Barney S., Hu G., Aurum A., Wohlin C.	Creating software product value in China	Journal	2009
Stober T., Hansmann U.	WebSphere portal 6.1: An agile development approach	Journal	2009

Anexo A. Información complementaria del capítulo 2

Autores	Título	Tipo de publicación	Año
Srivastava P.R., Pareek D.	Component prioritization schema for achieving maximum time and cost benefits from software testing	Conferencia	2009
Balikuddembe J.K., Potgieter A.E.	Using actor object operations structures to understand project requirements complexities	Conferencia	2008
Petersen K., Ronkko K., Wohlin C.	The impact of time controlled reading on software inspection effectiveness and efficiency: A controlled experiment	Conferencia	2008
Mohamed A.S.I., El-Maddah B.I.A., Wahba C.A.M.	Criteria-based requirements prioritization for software product management	Conferencia	2008
Beg R., Abbas Q., Verma R.P.	An approach for requirement prioritization using B-tree	Conferencia	2008
Brinkkemper S., Van De Weerd I., Saeki M., Versendaal J.	Process improvement in requirements management: A method engineering approach	Conferencia	2008
Barney S., Aurum A., Wohlin C.	A product management challenge: Creating software product value through requirements selection	Journal	2008
Some S.S.	Specifying use case sequencing constraints using description elements	Conferencia	2007
Kundu D., Samanta D.	A novel approach of prioritizing use case scenarios	Conferencia	2007
Morimoto S.	A system behavior analysis technique with visualization of a customer's domain	Conferencia	2007
Yip J.	Hands-on release planning with poker chips	Conferencia	2007
Li C., Van Den Akker J.M., Brinkkemper S., Diepen G.	Integrated requirement selection and scheduling for the release planning of a software product	Conferencia	2007
Laurent P., Cleland-Huang J., Duan C.	Towards automated requirements triage	Conferencia	2007
Perini A., Ricca F., Susi A., Bazzanella C.	An empirical study to compare the accuracy of AHP and CBRanking techniques for requirements prioritization	Conferencia	2007
Regnell B., Host M., Svensson R.B.	A quality performance model for cost-benefit analysis of non-functional requirements applied to the mobile handset domain	Conferencia	2007
Karlsson L., Thelin T., Regnell B., Berander P., Wohlin C.	Pair-wise comparisons versus planning game partitioning-experiments on requirements prioritisation techniques	Journal	2007
Azar J., Smith R.K., Cordes D.	Value-oriented requirements prioritization in a small development organization	Journal	2007
Gunes Koru A., Liu H.	Identifying and characterizing change-prone classes in two large-scale open-source products	Journal	2007
Peeters, J.; Dyson, P.	Cost-Effective Security	Journal	2007
McGrath O.G.	Balancing act: community and local requirements in an open source development process	Conferencia	2006
Baker P., Harman M., Steinhofel K., Skaliotis A.	Search based approaches to component selection and prioritization for the next release problem	Conferencia	2006
Berander P., Jonsson P.	Hierarchical Cumulative Voting (HCV) prioritization of requirements in hierarchies	Journal	2006
Li P.L., Herbsleb J., Shaw M., Robinson B.	Experiences and results from initiating field defect prediction and product test prioritization efforts at ABB Inc.	Conferencia	2006
Bercial P.J.L., Otero J.J.E., Sanz L.F., Gonzalez J.R.H.	A study of the relationship between usability and test cases precedence based on a formal model for activity diagrams	Conferencia	2005
Pereplechikov M., Padgham L.	Systematic incremental development of agent systems, using prometheus	Conferencia	2005
Ruhe G., Momoh J.	Strategic release planning and evaluation of operational feasibility	Conferencia	2005
Jeff Patton	Finding the forest in the trees	Conferencia	2005

Autores	Título	Tipo de publicación	Año
Losavio F., Chirinos L., Matteo A., Levy N., Ramdane-Cherif A.	Designing quality architecture: Incorporating ISO standards into the unified process	Journal	2004
Blotner J.A.	PIP: A product planning strategy for the whole family: Or how we became the brady bunch	Conferencia	2004
Greer D., Ruhe G.	Software release planning: An evolutionary and iterative approach	Journal	2004
Thelin, T; Andersson, P; Harrell, J	Tool support for usage-based reading	Conferencia	2004
Thelin T., Erlansson M., Host M., Wohlin C.	Experimentation with usage-based reading	Conferencia	2003
Thelin T., Runeson P., Wohlin C.	Prioritized use cases as a vehicle for software inspections	Journal	2003
Denger, C; Ciolkowski, M	High quality statecharts through tailored, perspective-based inspections	Conferencia	2003
Denger, C.; Ciolkowski, M.	High quality statecharts through tailored	Conferencia	2003
Olvingson C., Hallberg N., Timpka T., Lindqvist K.	Requirements Engineering for inter-organizational health information systems with functions for spatial analyses: Modeling a WHO safe community applying Use Case Maps	Journal	2002
Kim J.-M., Porter A.	A history-based test prioritization technique for regression testing in resource constrained environments	Conferencia	2002
Thelin, T; Runeson, P; Wohlin, C; Olsson, T; Andersson, C	How much information is needed for usage-based reading? A series of experiments	Conferencia	2002
Thelin T., Runeson P., Regnell B.	Usage-based reading - An experiment to guide reviewers with use cases	Journal	2001
Regnell B., Host M., Nattoch Dag J., Beremark P., Hjelm T.	An industrial case study on distributed prioritisation in market-driven requirements engineering for packaged software	Journal	2001
Carlshamre P., Sandahl K., Lindvall M., Regnell B., Natt och Dag J.	An industrial survey of requirements interdependencies in software product release planning	Conferencia	2001
Moisiadis Frank	Prioritizing use cases and scenarios	Journal	2000
Moisiadis Frank	Prioritizing scenario evolution	Conferencia	2000
Gruenbacher, P	Collaborative requirements negotiation with EasyWinWin	Conferencia	2000
Kim Youngchul, Carlson C.Robert	Scenario based integration testing for object-oriented software development	Journal	1999
Van Den Berg K.G., Simons A.J.H.	Control-flow semantics of use cases in UML	Journal	1999
Runeson Per, Regnell Bjorn	Derivation of an integrated operational profile and use case model	Conferencia	1998
Runeson, P.; Regnell, B.	Derivation of an integrated operational profile and use case model	Conferencia	1998
Ryan Kevin, Karlsson Joachim	Prioritizing software requirements in an industrial setting	Conferencia	1997

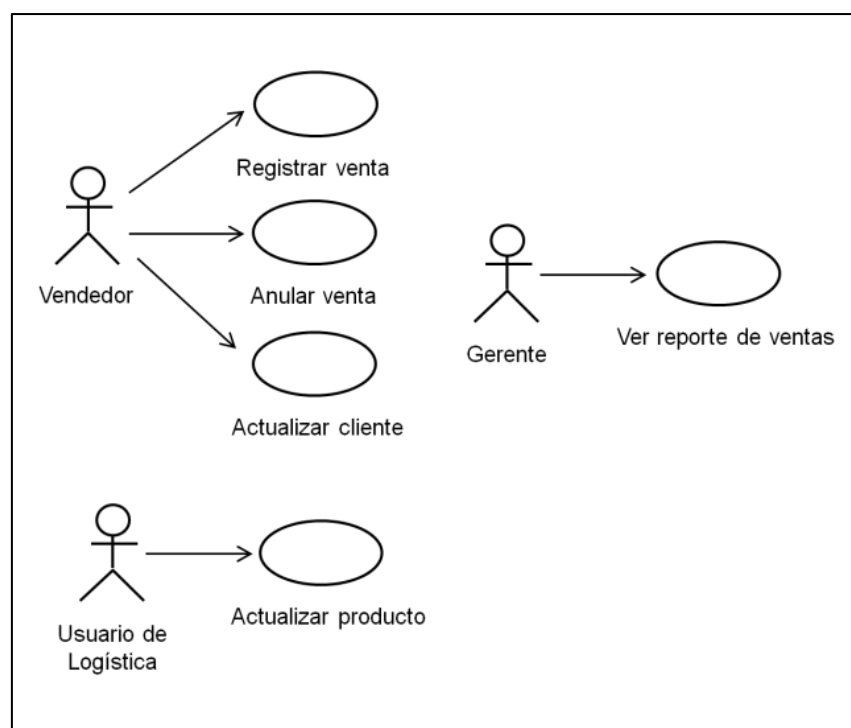
Información Complementaria del Capítulo 4

Anexo

B

En este anexo se incluye información complementaria del capítulo 4 que corresponde a las especificaciones de casos de uso del ejemplo que se utiliza en dicho capítulo. La sección B1 muestra el diagrama de casos de uso y desde las secciones B2 hasta la B6 presentan las especificaciones de casos de uso.

B.1 Diagramas de Casos de Uso



B.2 Actualizar Producto

Descripción: El propósito del caso de uso es actualizar la información de los productos; es decir, crear, modificar o eliminar los productos que vende la empresa

Precondición: No existen para este caso de uso

Poscondiciones: Los datos del o los productos han sido actualizados.

Flujo Básico

1. El caso de uso empieza cuando el usuario elige la opción “Administrar Productos” opción “Agregar producto”.
2. El sistema muestra un formulario con los siguientes datos: código, nombre y precio.
3. El usuario ingresa los valores correspondientes al producto y selecciona “Aceptar”.
4. El sistema guarda la información ingresada, y el caso de uso finaliza.

Flujo Alternativo: Modificar información de producto

1. El actor selecciona el “Modificar información de producto”.
2. El sistema muestra una relación de los productos existentes: código y nombre.
3. El usuario escoge uno de los productos de la lista y selecciona “Modificar”.
4. El sistema carga nuevamente el formulario con los datos del producto a editar (código, nombre y precio).
5. El usuario modifica los valores correspondientes a las opciones y selecciona “Aceptar”.
6. El sistema guarda la información ingresada, y el caso de uso finaliza.

Flujo Alternativo: Eliminar producto

1. El actor selecciona el “Eliminar producto”.
2. El sistema muestra una relación de los productos existentes: código y nombre.
3. El usuario selecciona uno de los productos de la lista y selecciona “Eliminar”.
4. El sistema muestra un mensaje de confirmación de la eliminación.
5. El usuario selecciona “Sí”.
6. El sistema guarda la información ingresada, y el caso de uso finaliza.

Flujo Excepcional

El flujo excepcional se da en el paso 3 del flujo básico, el paso 5 de los flujos alternativos o cuando el usuario escoge la opción cancelar, el sistema pedirá una confirmación. De darse ésta, el caso de uso finaliza.

B.3 Actualizar Cliente

Descripción: El propósito del caso de uso es actualizar la información de los clientes; es decir, crear, modificar o eliminar los clientes de la empresa.

Precondiciones: No existen para este caso de uso

Poscondiciones: Los datos del o los clientes han sido actualizados

Flujo Básico

1. El caso de uso empieza cuando el usuario elige la opción “Actualizar clientes” opción “Agregar clientes”.
2. El sistema muestra un formulario con los siguientes datos: código, y nombre.
3. El usuario ingresa los valores correspondientes al cliente y selecciona “Aceptar”.
4. El sistema guarda la información ingresada, y el caso de uso finaliza.

Flujo Alternativo: Modificar información de cliente

1. El actor selecciona el “Modificar información de cliente”.
2. El sistema muestra una relación de los clientes de la empresa: código y nombre.
3. El usuario escoge uno de los productos de la lista y selecciona “Modificar”.
4. El sistema carga nuevamente el formulario con los datos del cliente a editar (código y nombre).
5. El usuario modifica los valores correspondientes a las opciones y selecciona “Aceptar”.
6. El sistema guarda la información ingresada, y el caso de uso finaliza.

Flujo Alternativo: Eliminar producto

1. El actor selecciona el “Eliminar cliente”.
2. El sistema muestra una relación de los productos existentes: código y nombre.
3. El usuario selecciona uno de los clientes de la lista y selecciona “Eliminar”.
4. El sistema muestra un mensaje de confirmación de la eliminación.
5. El usuario selecciona “Sí”.
6. El sistema guarda la información ingresada, y el caso de uso finaliza.

Flujo Excepcional

El flujo excepcional se da en el paso 3 del flujo básico, el paso 5 de los flujos alternativos o cuando el usuario escoge la opción cancelar, el sistema pedirá una confirmación. De darse ésta, el caso de uso finaliza.

B.4 Registrar Venta

Descripción: El propósito del caso de uso es registrar la venta de productos

Precondiciones: No existen para este caso de uso

Poscondiciones: Se ha registrado una venta de productos

Flujo Básico

1. El caso de uso empieza cuando el usuario elige la opción “Registra venta”.
2. El sistema muestra un formulario donde solicita al usuario ingresar los productos a vender y sus respectivas cantidades.

3. El usuario registra uno de los productos de la venta por su código.
4. El sistema muestra la descripción del producto, el precio, subtotal y el total calculado.
5. Los pasos 3 y 4 son repetidos para cada producto que el actor ingrese.
6. El usuario selecciona "Terminar"
7. El sistema guarda los datos de la venta, asigna un número de factura, genera el documento de venta que la certifica y el caso de uso finaliza.

Flujo Excepcional

El flujo excepcional se da en los pasos 3, 4 o 5 del flujo básico cuando el usuario quiere cancelar la operación, el sistema pedirá una confirmación. De darse ésta, el caso de uso finaliza.

B.5 Anular Venta

Descripción: El propósito del caso de uso es anular una venta de productos

Precondiciones: No existen para este caso de uso

Poscondiciones: Se ha anulado una venta

Flujo Básico

1. El caso de uso empieza cuando el usuario elige la opción "Anular venta".
2. El sistema muestra un formulario donde solicita al usuario ingresar el número de factura.
3. El usuario ingresa el número de factura.
4. El sistema muestra la información de la venta: número, fecha, total. Para el detalle de la venta muestra código del producto, nombre del producto y subtotal.
5. El usuario selecciona anular.
6. El sistema guarda los datos de la anulación de la venta y el caso de uso finaliza.

Flujos Excepcionales

Si el número de factura no existe (paso 3 del flujo básico) el sistema mostrará un mensaje de error y permitirá al usuario ingresar otro número de factura.

En el paso 5 del flujo básico cuando el usuario quiere cancelar la operación, el sistema pedirá una confirmación. De darse ésta, el caso de uso finaliza.

B.6 Ver Reporte de Ventas

Descripción: El propósito del caso de uso es mostrar un reporte de ventas de los productos

Precondiciones: No existen para este caso de uso

Poscondiciones: Se ha mostrado el reporte de ventas

Flujo Básico

1. El caso de uso empieza cuando el usuario elige la opción “Ver reporte de ventas”.
2. El sistema muestra una ventana con las ventas realizadas por el empleado: código de empleado, nombre del empleado y total vendido. Además, muestra por producto el total que se vendió, para ello presenta la siguiente información: código del producto, nombre del producto y total vendido. Al finalizar muestra el monto total facturado.
3. El usuario selecciona salir y el caso de uso finaliza.

<p style="text-align: center;">Información Complementaria del Capítulo 5</p>	Anexo
	C

En este anexo se incluye información complementaria relacionada a la evaluación de la conversión de ficheros de UML₂FP (ver capítulo 5). La Sección C.1 contiene información relacionada al experimento controlado realizado con profesionales EC1-PROF; la Sección C.1, información asociada al estudio piloto realizado con docentes y asistentes de docencia EP2-DOC; la Sección C.2, información relacionada al experimento realizado con estudiantes de pregrado EC3-PREG; la Sección C.4 y la Sección C.5, información relacionada al estudio con estudiantes de posgrado EC4-POST y EC5-POST respectivamente.

C.1 Evaluación con Profesionales: EC1-PROF

En esta sección se presenta la información complementaria del estudio realizado con profesionales EC1-PROF. En la Subsección C.1.1, se muestran los casos de estudio que fueron empleados por los profesionales; en la Subsección C.1.2, se indican los formularios y cuestionario que tuvieron que completar los participantes; en la Subsección C.1.3, se presentan tablas con la información obtenida de los formularios y cuestionarios; y, finalmente, en la Subsección C.1.4, se incluyen los formularios y el cuestionario que fueron llenados por un participante de la evaluación, el cual se incluye como ejemplo.

Las imágenes escaneadas de los formularios y cuestionario llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

C.1.1 Casos

Las figuras que se muestran a continuación presentan los casos que fueron entregados a los participantes del estudio. Las tres primeras figuras (Figura C-1, Figura C-2 y Figura C-2) muestran las páginas de los casos con diagramas Entidad-Relación. Las tres últimas figuras de esta sección (Figura C-4, Figura C-5 y Figura C-6) presentan las páginas de los casos con diagramas de clases.

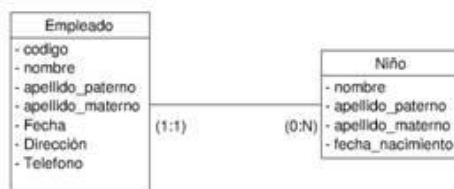
Casos con diagramas E-R

Casos para la Identificación de Ficheros en Puntos de Función con Diagramas Entidad-Relación

Situación 1

Una aplicación de Recursos Humanos mantiene la información de los empleados y sus hijos. Es política de la empresa no contratar a pareja de esposos, es decir, sólo podría trabajar el padre o la madre, pero no ambos.

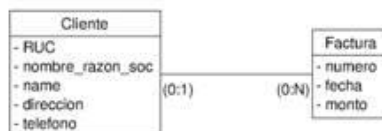
La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha de nacimiento, dirección y teléfono. La información del hijo es la siguiente: nombre, apellido paterno, apellido materno y fecha de nacimiento.



Situación 2

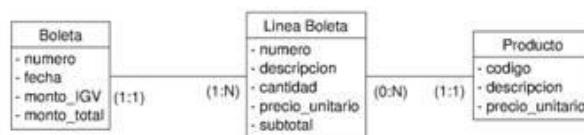
En un sistema de ventas, sólo se quiere automatizar lo siguiente: el registro de clientes y el registro de los datos principales de la factura de cada uno de los clientes. La información del cliente es la siguiente: RUC, nombre o razón social, dirección y teléfono. La información de la factura es: número, fecha y monto total.

En este caso, pueden existir clientes sin facturas y facturas sin clientes.



Situación 3

Se quiere realizar un sistema que administrará boletas de venta. La información de las boletas es: número, fecha, monto IGV, monto total. Adicionalmente cada línea de la boleta (detalle de la boleta) contendrá: número de línea, descripción del producto, cantidad del producto, precio unitario del producto y subtotal. El sistema también debe permitir el registro de los productos que vende, cuya información es: código, descripción y precio unitario.

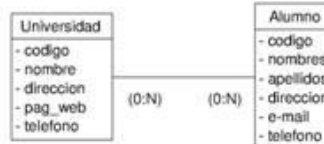


José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura C-1. Caso con diagramas E-R para EC1-PROF - Pág. 1/3

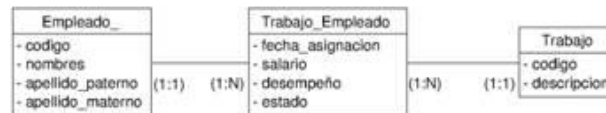
Situación 4

Se requiere implementar un sistema que registre las universidades y sus alumnos. La información de cada universidad es: código, nombre, dirección, página Web y teléfono. La información de un alumno es: código, nombres y apellidos, dirección, E-mail y teléfono. El sistema deberá permitir que las universidades no tengan alumnos registrados y que los alumnos no pertenezcan a ninguna universidad. Los alumnos pueden pertenecer a más de una universidad.



Situación 5

Una aplicación de Recursos Humanos mantiene la información de los empleados (código, nombre, apellido paterno y apellido materno) y de los trabajos (código y descripción). Además, mantiene la información de los trabajos asignados a un trabajador (fecha de asignación, salario, desempeño y estado). La información de asignación de trabajo deberá ser mantenida aún si el empleado ya no está asociado a un trabajo o si el trabajo ya se ha desactivado.



Situación 6

Se requiere elaborar un sistema que permita registrar y mantener la información de CDs. La información del CD tiene lo siguiente: código, título, duración, género. Por cada pista o "track" del CD, debería registrarse lo siguiente: número de pista, nombre de canción, nombre de artista y duración.

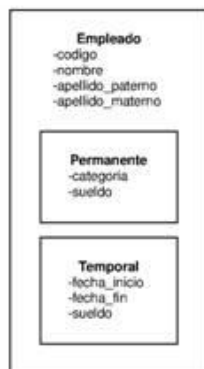


José Antonio Pino-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-2. Caso con diagramas E-R para EC1-PROF - Pág. 2/3

Situación 7

Una aplicación de Recursos Humanos mantiene la información de los empleados (código, nombre, apellido paterno y apellido materno). Adicionalmente, hay dos tipos de empleados: Permanente y Temporal. Los empleados permanentes tienen la siguiente información adicional: categoría y sueldo. Los empleados temporales tienen la siguiente información adicional: fecha de inicio, fecha de fin y sueldo.



José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-3. Caso con diagramas E-R para EC1-PROF - Pág. 3/3

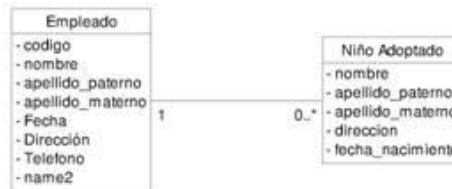
Caso con diagramas de clases

Casos para la Identificación de Ficheros en Puntos de Función para Diagrama de Clases de Análisis

Situación 1

Una compañía decide realizar un programa de “adopción” de niños, de manera que un trabajador pueda “adoptar” la cantidad de niños que quiera para darles juguetes en Navidad y en su cumpleaños. Un niño no puede ser adoptado por más de un trabajador y todos los niños siempre deberán tener un padre o madre adoptivo.

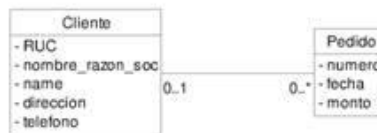
La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha de nacimiento, dirección y teléfono. La información del niño adoptado es la siguiente: nombre, apellido paterno, apellido materno, dirección y fecha de nacimiento.



Situación 2

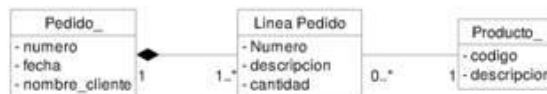
En un sistema de ventas, sólo se quiere automatizar lo siguiente: el registro de clientes y el registro de los datos principales de un pedido de cada uno de los clientes. La información del cliente es la siguiente: RUC, nombre o razón social, dirección y teléfono. La información del pedido es: número, fecha y monto total del pedido.

En este caso, pueden existir clientes sin pedidos y pedidos sin clientes.



Situación 3

Se quiere realizar un sistema que permita el registro de pedidos de los clientes. La información de los pedidos es: número, fecha y nombre del cliente. Adicionalmente cada línea del pedido (detalle del pedido) contendrá: número de línea, descripción del producto y cantidad del producto. El sistema también debe permitir el registro de los tipos de productos que vende, cuya información es: código y descripción.

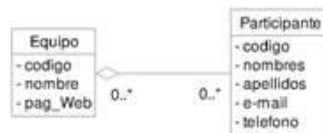


José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura C-4. Caso con diagramas de clases para EC1-PROF - Pág. 1/3

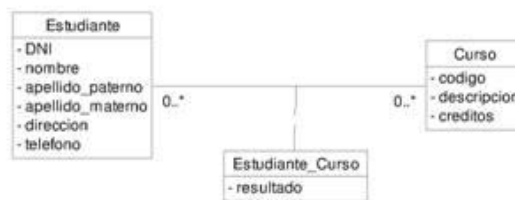
Situación 4

Para los trabajos de curso de una universidad se requiere hacer un sistema que registre los Equipos de Proyecto y los participantes de los equipos. La información de los Equipos es: código, nombre y página Web. La información de los participantes es: código, nombres, apellidos, E-mail y teléfono. El sistema deberá permitir que los equipos no tengan miembros registrados y que los participantes no pertenezcan a un equipo. Los participantes podrían pertenecer a más de un equipo.



Situación 5

Para el sistema de una institución educativa, se mantiene la información de los estudiantes (DNI, nombre, apellido paterno, apellido materno, dirección y teléfono) y de los cursos (código, descripción, número de créditos). Además, se mantiene la información de quien tomó un curso y el resultado obtenido. La información de quien tomó el curso deberá ser mantenida aún si el alumno ya no está asociado a un curso o si el curso ya no se dicta.



Situación 6

Se requiere elaborar un sistema que permita registrar y mantener la información de CDs. La información del CD tiene lo siguiente: código, título, duración, género. Por cada pista o "track" del CD, debería registrarse lo siguiente: número de pista, nombre de canción, nombre de artista y duración.



José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura C-5. Caso con diagramas de clases para EC1-PROF - Pág. 2/3

Situación 7

La aplicación de una institución educativa mantiene la información de los miembros de la institución (código, nombre, apellido paterno y apellido materno). Adicionalmente, hay dos tipos de miembros: profesores y alumnos. Los profesores tienen la siguiente información adicional: sueldo y categoría. Los alumnos tienen la siguiente información adicional: fecha de ingreso y rendimiento académico.



José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-8051

Figura C-6. Caso con diagramas de clases para EC1-PROF - Pág. 3/3

C.1.2 Formularios y cuestionario

La Figura C-7 y la Figura C-8 muestran los formularios que tuvieron que completar los participantes del estudio. El primero de ellos fue utilizado por los participantes para registrar los resultados de aplicar la técnica del Ifpug y el segundo, los resultados de aplicar UML₂FP. La Figura C-9 muestra el cuestionario en el que los participantes tuvieron que registrar sus percepciones sobre las técnicas.

Resultados con Técnica para Entidad-Relación

Nombres y apellidos: _____

Indique la hora de inicio: _____

Sit.	Nombre Fichero	Nombre de DETs	Núm. DETs	Núm. RET	PFsA
1					
2					
3					
4					
5					
6					
7					

Indique la hora de fin: _____

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-R01

Figura C-7. Formulario para Entidad-Relación

Resultados con Técnica para Diagrama de Clases de Análisis

Nombres y apellidos: _____

Indique la hora de inicio: _____

Sit.	Nombre Fichero	Nombre de DETs	Núm. DETs	Núm. RET	PFsA
1					
2					
3					
4					
5					
6					
7					

Indique la hora de fin: _____

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-8. Formulario para diagramas de clases

Cuestionario Final - PE

Nombre y apellidos: _____

1. ¿Cuánto tiempo de experiencia en años tiene en desarrollo de software? _____
2. ¿Sabía de la existencia de la técnica de puntos de función? _____
3. ¿Ha utilizado la técnica de puntos de función en proyectos de su empresa? _____ ¿En cuántos proyectos? _____
4. Antes de esta experimentación ¿Sabía hacer un diagrama E-R? _____
5. Antes de esta experimentación ¿Sabía hacer un diagrama de clases de análisis? _____
6. ¿Cómo evaluaría su conocimiento sobre diagramas E-R antes de esta experimentación?
a) Malo b) Regular c) Bueno d) Muy bueno
7. ¿Cómo evaluaría su conocimiento sobre diagramas diagrama de clases de análisis antes de esta experimentación?
a) Malo b) Regular c) Bueno d) Muy bueno
8. ¿Con cuánta frecuencia se utilizan diagramas E-R en los proyectos en los que ha trabajado o trabaja?
a) Poco b) Aproximadamente en la mitad de proyectos c) Casi siempre d) Siempre
9. ¿Con cuánta frecuencia se utilizan diagramas de clases de análisis en los proyectos en los que ha trabajado o trabaja?
a) Poco b) Aproximadamente en la mitad de proyectos c) Casi siempre d) Siempre
10. ¿Cuál de las dos técnicas le parece que es más clara de aplicar?
a) La que emplea E-R b) La que emplea clases c) Las dos son iguales

¿Por qué?

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura C-9. Cuestionario empleado para EC1-PROF

C.1.3 Tablas con información obtenida de los formularios y del cuestionario

La Tabla C-1 y la Tabla C-2 muestran los resultados obtenidos en los formularios que completaron los participantes al experimento controlado. Un valor de “1” indica que identificó correctamente los ficheros y sus RET y “0” que no lo identificó de manera correcta.

Tabla C-1. Resultados obtenidos en el formulario E-R de EC1-PROF

Participante	Caso						
	1	2	3	4	5	6	7
1	0	1	0	1	1	1	1
2	0	1	0	1	1	1	1
3	1	1	1	1	0	1	1
4	1	1	1	1	0	1	0
5	0	1	1	1	0	1	0
6	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1
8	1	1	1	1	0	1	1
9	1	1	1	1	1	1	1
10	1	1	0	1	0	1	1
11	0	1	1	1	1	1	1
12	0	1	1	1	1	1	1
13	0	1	1	1	0	1	1
14	0	1	1	1	1	1	0
15	1	1	1	1	1	1	1
16	1	1	1	1	1	1	1

Tabla C-2. Resultados obtenidos en el formulario sobre diagramas de clases de EC1-PROF

Participante	Caso						
	1	2	3	4	5	6	7
1	0	1	0	1	1	1	1
2	0	0	1	0	0	1	0
3	0	1	1	1	1	1	0
4	0	1	1	1	0	1	1
5	0	1	1	1	1	1	1
6	0	1	1	1	1	1	1
7	1	1	1	1	1	1	1
8	0	1	1	1	0	1	1

Participante	Caso						
	1	2	3	4	5	6	7
9	0	1	1	1	1	1	1
10	0	1	1	1	1	1	1
11	0	0	1	1	1	1	1
12	1	1	1	1	1	1	1
13	0	1	1	1	1	1	1
14	1	1	1	1	1	1	0
15	0	1	1	1	1	1	0
16	0	1	1	1	1	1	1

La Tabla C-3 muestra los resultados que se obtuvieron en la pregunta 10 del cuestionario. Esta pregunta corresponde a seleccionar cuál de las dos técnicas es más clara de aplicar. La respuesta a) corresponde a E-R, la respuesta b) a diagramas de clases y c) que las dos son iguales.

Tabla C-3. Resultados obtenidos del cuestionario de EC1-PROF para la pregunta 10

Participante	Respuesta Pregunta 10
1	c
2	c
3	a
4	a
5	b
6	c
7	b
8	a
9	b
10	b
11	c
12	b
13	a
14	b
15	b
16	c

C.1.4 Cuestionarios llenados y diagramas realizados por un participante del estudio

En la Figura C-10, la Figura C-11 y la Figura C-12, se muestran como ejemplo los formularios y cuestionario llenados por uno de los profesionales que participaron en el estudio. Las imágenes escaneadas de los formularios y cuestionarios llenados por todos los participantes se encuentran en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

Resultados con Técnica para Entidad-Relación

Nombres y apellidos: Luz Tarazona Cruz

Indique la hora de inicio: 20:09

Sit.	Nombre Fichero	Nombre de DETs	FTR	Núm. DETs	Núm. RET	PFsA ILF
1	Empleado- Empleado		1	127	2	7PFA
	Niño		2	5	1	
2	Ciente		2	4	1	7PFA
	Factura			34	1	
3	Bolita Bolita			41	1	7PFA
	Línea Bolita			641	2	
	Producto			3	1	7PFA
4	Universidad		1	6	1	
	Alumno		1	7	1	
5	Empleado			4	1	7PFA
	Trobo Emp			6	2	
	Troboap			2	1	
6	CD- Pista			9	2	
7	Empleado			9	2	7PFA

Indique la hora de fin: 20:50

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú. Proyecto DAI-4051

Figura C-10. Formulario para Entidad-Relación del participante 1 de EC1-PROF

Resultados con Técnica para Diagrama de Clases de Análisis

Nombres y apellidos: Luz Talarzonca Cruz

Indique la hora de inicio: 21:00

Sit.	Nombre Fichero	Nombre de DETs	Núm. DETs	Núm. RET	PFsA
1	Empleado		7	1	7
			6	1	
2	Cliente		4	1	7
	Pedido		4	1	
3	Pedido Pedido		3	1	7
	Línea Pedido		4	2	
	Producto		2	1	
4	Equipo		4	1	7
	Participante		6	1	
5	Estudiante		7	1	7
	Curso		4	2	
	Estudiante Curso		1	1	
6	CD-Pista		9	2	7
7	Miembro		4	1	7

Indique la hora de fin: 21:20

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-11. Formulario para diagramas de clases del participante 1 de EC1-PROF

Cuestionario Final - PF

Nombre y apellidos: Luz Tarazona Cruz

1. ¿Cuánto tiempo de experiencia en años tiene en desarrollo de software? 03
2. ¿Sabía de la existencia de la técnica de puntos de función? Si
3. ¿Ha utilizado la técnica de puntos de función en proyectos de su empresa? No ¿En cuántos proyectos? _____
4. Antes de esta experimentación ¿Sabía hacer un diagrama E-R? Si
5. Antes de esta experimentación ¿Sabía hacer un diagrama de clases de análisis? No
6. ¿Cómo evaluaría su conocimiento sobre diagramas E-R antes de esta experimentación?

a) Malo	<input checked="" type="radio"/> b) Regular	c) Bueno	d) Muy bueno
---------	---	----------	--------------
7. ¿Cómo evaluaría su conocimiento sobre diagramas diagrama de clases de análisis antes de esta experimentación?

a) Malo	<input checked="" type="radio"/> b) Regular	c) Bueno	d) Muy bueno
---------	---	----------	--------------
8. ¿Con cuánta frecuencia se utilizan diagramas E-R en los proyectos en los que ha trabajado o trabaja?

<input checked="" type="radio"/> a) Poco	b) Aproximadamente en la mitad de proyectos	c) Casi siempre	d) Siempre
--	---	-----------------	------------
9. ¿Con cuánta frecuencia se utilizan diagramas de clases de análisis en los proyectos en los que ha trabajado o trabaja?

<input checked="" type="radio"/> a) Poco	b) Aproximadamente en la mitad de proyectos	c) Casi siempre	d) Siempre
--	---	-----------------	------------
10. ¿Cuál de las dos técnicas le parece que es más clara de aplicar?

a) La que emplea E-R	b) La que emplea clases	<input checked="" type="radio"/> c) Las dos son iguales
----------------------	-------------------------	---

¿Por qué?

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DM-4031

Figura C-12. Cuestionario del participante 1 de EC1-PROF

C.2 Evaluación con Docentes: EP2-DOC

Esta sección presenta la información complementaria del estudio piloto realizado con docentes EP2-DOC. Dado que los formularios y cuestionario empleados fueron los mismos que los utilizados en el estudio EC1-PROF, se incluye en esta sección los casos que usaron los participantes (ver Subsección C.2.1) y las tablas con la información obtenida de los formularios y cuestionario obtenidos (ver Subsección C.2.2).

Las imágenes escaneadas de los cuestionarios llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

C.2.1 Casos

Las figuras que se muestran a continuación presentan los casos que fueron entregados a los participantes del estudio. Las tres primeras figuras (Figura C-14, Figura C-15 y Figura C-16) muestran las páginas de los casos con diagramas Entidad-Relación. Las tres figuras últimas de esta sección (Figura C-17, Figura C-18 y Figura C-19) presentan las páginas de los casos con diagramas de clases.

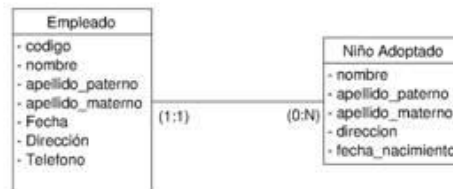
Casos con diagramas E-R

Casos para la Identificación de Ficheros en Puntos de Función para Diagrama Entidad Relación

Situación 1

Una compañía decide realizar un programa de “adopción” de niños, de manera que un trabajador pueda “adoptar” la cantidad de niños que quiera para darles juguetes en Navidad y en su cumpleaños. Un niño no puede ser adoptado por más de un trabajador y todos los niños siempre deberán tener un padre o madre adoptivo.

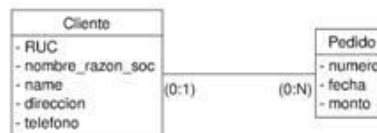
La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha de nacimiento, dirección y teléfono. La información del niño adoptado es la siguiente: nombre, apellido paterno, apellido materno, dirección y fecha de nacimiento.



Situación 2

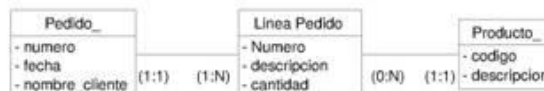
En un sistema de ventas, sólo se quiere automatizar lo siguiente: el registro de clientes y el registro de los datos principales de un pedido de cada uno de los clientes. La información del cliente es la siguiente: RUC, nombre o razón social, dirección y teléfono. La información del pedido es: número, fecha y monto total del pedido.

En este caso, pueden existir clientes sin pedidos y pedidos sin clientes.



Situación 3

Se quiere realizar un sistema que permita el registro de pedidos de los clientes. La información de los pedidos es: número, fecha y nombre del cliente. Adicionalmente cada línea del pedido (detalle del pedido) contendrá: número de línea, descripción del producto y cantidad del producto. El sistema también debe permitir el registro de los tipos de productos que vende, cuya información es: código y descripción.

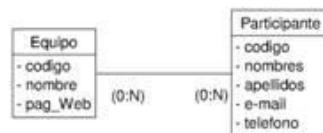


José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura C-14. Caso con diagramas E-R para EP2-DOC - Pág. 1/3

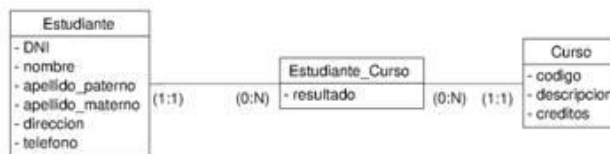
Situación 4

Para los trabajos de curso de una universidad se requiere hacer un sistema que registre los Equipos de Proyecto y los participantes de los equipos. La información de los Equipos es: código, nombre y página Web. La información de los participantes es: código, nombres, apellidos, E-mail y teléfono. El sistema deberá permitir que los equipos no tengan miembros registrados y que los participantes no pertenezcan a un equipo. Los participantes podrían pertenecer a más de un equipo.



Situación 5

Para el sistema de una institución educativa, se mantiene la información de los estudiantes (DNI, nombre, apellido paterno, apellido materno, dirección y teléfono) y de los cursos (código, descripción, número de créditos). Además, se mantiene la información de quien tomó un curso y el resultado obtenido. La información de quien tomó el curso deberá ser mantenida aún si el alumno ya no está asociado a un curso o si el curso ya no se dicta.



Situación 6

Se requiere elaborar un sistema que permita registrar y mantener la información de CDs. La información del CD tiene lo siguiente: código, título, duración, género. Por cada pista o "track" del CD, debería registrarse lo siguiente: número de pista, nombre de canción, nombre de artista y duración.

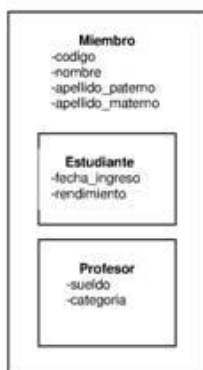


José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-15. Caso con diagramas E-R para EP2-DOC - Pág. 2/3

Situación 7

La aplicación de una institución educativa mantiene la información de los miembros de la institución (código, nombre, apellido paterno y apellido materno). Adicionalmente, hay dos tipos de miembros: profesores y alumnos. Los profesores tienen la siguiente información adicional: sueldo y categoría. Los alumnos tienen la siguiente información adicional: fecha de ingreso y rendimiento académico.



José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-8051

Figura C-16. Caso con diagramas E-R para EP2-DOC - Pág. 3/3

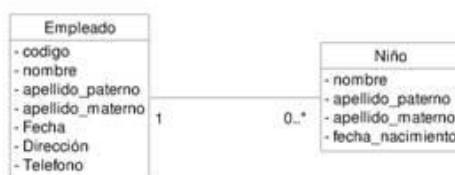
Caso con diagramas de clases

Casos para la Identificación de Ficheros en Puntos de Función con Diagramas de Clases

Situación 1

Una aplicación de Recursos Humanos mantiene la información de los empleados y sus hijos. Es política de la empresa no contratar a pareja de esposos, es decir, sólo podría trabajar el padre o la madre, pero no ambos.

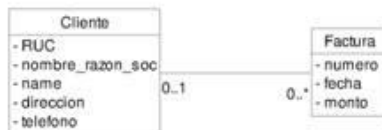
La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha de nacimiento, dirección y teléfono. La información del hijo es la siguiente: nombre, apellido paterno, apellido materno y fecha de nacimiento.



Situación 2

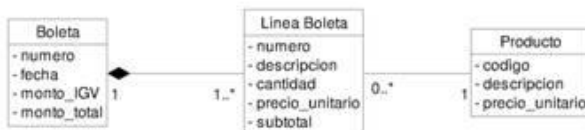
En un sistema de ventas, sólo se quiere automatizar lo siguiente: el registro de clientes y el registro de los datos principales de la factura de cada uno de los clientes. La información del cliente es la siguiente: RUC, nombre o razón social, dirección y teléfono. La información de la factura es: número, fecha y monto total.

En este caso, pueden existir clientes sin facturas y facturas sin clientes.



Situación 3

Se quiere realizar un sistema que administrará boletas de venta. La información de las boletas es: número, fecha, monto IGV, monto total. Adicionalmente cada línea de la boleta (detalle de la boleta) contendrá: número de línea, descripción del producto, cantidad del producto, precio unitario del producto y subtotal. El sistema también debe permitir el registro de los productos que vende, cuya información es: código, descripción y precio unitario.

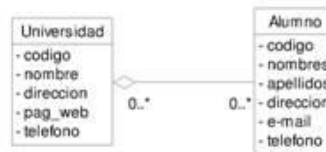


José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-17. Caso con diagramas de clases para EP2-DOC - Pág. 1/3

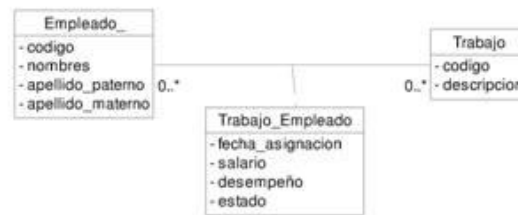
Situación 4

Se requiere implementar un sistema que registre las universidades y sus alumnos. La información de cada universidad es: código, nombre, dirección, página Web y teléfono. La información de un alumno es: código, nombres y apellidos, dirección, E-mail y teléfono. El sistema deberá permitir que las universidades no tengan alumnos registrados y que los alumnos no pertenezcan a ninguna universidad. Los alumnos pueden pertenecer a más de una universidad.



Situación 5

Una aplicación de Recursos Humanos mantiene la información de los empleados (código, nombre, apellido paterno y apellido materno) y de los trabajos (código y descripción). Además, mantiene la información de los trabajos asignados a un trabajador (fecha de asignación, salario, desempeño y estado). La información de asignación de trabajo deberá ser mantenida aún si el empleado ya no está asociado a un trabajo o si el trabajo ya se ha desactivado.



Situación 6

Se requiere elaborar un sistema que permita registrar y mantener la información de CDs. La información del CD tiene lo siguiente: código, título, duración, género. Por cada pista o "track" del CD, debería registrarse lo siguiente: número de pista, nombre de canción, nombre de artista y duración.

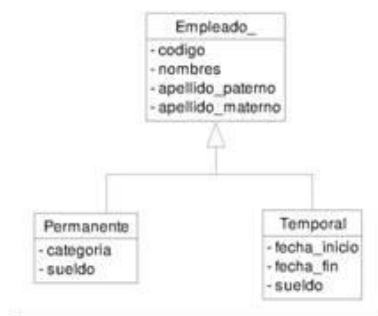


José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura C-18. Caso con diagramas de clases para EP2-DOC - Pág. 2/3

Situación 7

Una aplicación de Recursos Humanos mantiene la información de los empleados (código, nombre, apellido paterno y apellido materno). Adicionalmente, hay dos tipos de empleados: Permanente y Temporal. Los empleados permanentes tienen la siguiente información adicional: categoría y sueldo. Los empleados temporales tienen la siguiente información adicional: fecha de inicio, fecha de fin y sueldo.



José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-19. Caso con diagramas de clases para EP2-DOC - Pág. 3/3

C.2.2 Tablas con información obtenida de los formularios y del cuestionario

La Tabla C-4 y la Tabla C-5 muestran los resultados obtenidos en los formularios que completaron los participantes al experimento controlado. Un valor de “1” indica que identificó correctamente los ficheros y sus RET y “0” que no lo identificó de manera correcta.

Tabla C-4. Resultados obtenidos en el formulario E-R de EP2-DOC

Participante	Caso						
	1	2	3	4	5	6	7
1	0	1	1	1	1	1	0
2	0	1	1	1	1	1	1
3	1	1	1	1	1	1	1
4	0	1	1	1	1	1	1

Tabla C-5. Resultados obtenidos en el formulario sobre diagramas de clases de EP2-DOC

Participante	Caso						
	1	2	3	4	5	6	7
1	1	1	1	1	1	1	0
2	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1

C.3 Evaluación con Estudiantes de Pregrado: EC3-PREG

Esta sección presenta la información complementaria del estudio realizado con estudiantes de pregrado EC3-PREG. Dado que los formularios empleados fueron los mismos que los utilizados en el estudio EC1-PROF, no se incluyen en esta sección.

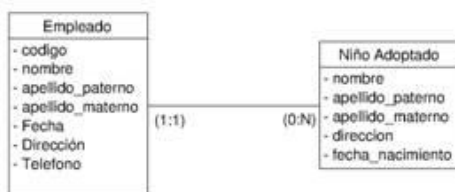
Según los resultados obtenidos en el estudio realizado con los docentes, se pudo determinar que al primer caso con diagramas Entidad-Relación le faltaba información. Por ello, se tuvo que modificar el caso en mención, manteniéndose sin cambios los casos restantes. La Figura C-20 muestra la primera página de los casos de E-R que fue modificada. No se incluyen las otras páginas ni los casos con diagramas de clases, porque estos no se modificaron (ver Subsección C.2.1 para ver el resto de páginas y los casos con diagramas de clases).

Casos para la Identificación de Ficheros en Puntos de Función para Diagrama Entidad Relación

Situación 1

Una compañía decide realizar un programa de “adopción” de niños, de manera que un trabajador pueda “adoptar” la cantidad de niños que quiera para darles juguetes en Navidad y en su cumpleaños. Un niño no puede ser adoptado por más de un trabajador y todos los niños siempre deberán tener un padre o madre adoptivo. Si a un empleado se le retira de la empresa, el o los niños que fueron adoptados por ese empleado son asignados a otra persona.

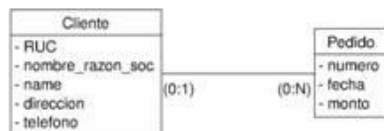
La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha de nacimiento, dirección y teléfono. La información del niño adoptado es la siguiente: nombre, apellido paterno, apellido materno, dirección y fecha de nacimiento.



Situación 2

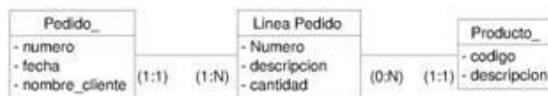
En un sistema de ventas, sólo se quiere automatizar lo siguiente: el registro de clientes y el registro de los datos principales de un pedido de cada uno de los clientes. La información del cliente es la siguiente: RUC, nombre o razón social, dirección y teléfono. La información del pedido es: número, fecha y monto total del pedido.

En este caso, pueden existir clientes sin pedidos y pedidos sin clientes.



Situación 3

Se quiere realizar un sistema que permita el registro de pedidos de los clientes. La información de los pedidos es: número, fecha y nombre del cliente. Adicionalmente cada línea del pedido (detalle del pedido) contendrá: número de línea, descripción del producto y cantidad del producto. El sistema también debe permitir el registro de los tipos de productos que vende, cuya información es: código y descripción.



José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-8051

Figura C-20. Caso con diagramas E-R para EC3-PREG - Pág. 1/3

La Figura C-21 muestra el cuestionario que fue empleado para el estudio con EC3-PREG. Este cuestionario es una simplificación del cuestionario empleado en EC1-PROF.

Cuestionario Final - PF

Nombre y apellidos: _____

I. ¿Cuál de las dos técnicas le parece que es más clara de aplicar?

a) La que emplea E-R b) La que emplea clases c) Las dos son iguales

¿Por qué?

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAI-4051

Figura C-21. Cuestionario empleado para EC3-PREG

La Tabla C-6 y la Tabla C-7 muestran los resultados obtenidos en los formularios que completaron los participantes al experimento controlado. Un valor de “1” indica que identificó correctamente los ficheros y sus RET y “0” que no lo identificó de manera correcta.

Tabla C-6. Resultados obtenidos en el formulario E-R de EC3-PREG

Participante	Caso						
	1	2	3	4	5	6	7
1	1	1	0	1	0	0	0
2	1	1	0	1	1	1	1
3	0	1	0	1	0	0	1
4	1	1	0	1	0	0	1
5	0	1	0	1	0	1	0
6	1	1	0	1	0	0	1
7	0	1	0	NR	0	1	0
8	0	1	1	1	1	1	1
9	1	1	0	0	0	0	NR
10	1	1	0	1	1	0	0
11	1	1	0	1	1	0	0
12	0	1	1	1	1	1	1
13	1	1	0	1	1	0	0
14	1	1	0	1	1	1	1
15	0	1	0	1	0	1	0
16	1	1	0	1	0	0	1
17	0	1	0	1	1	1	1
18	1	1	0	1	1	0	1
19	1	1	1	1	1	1	1
20	0	1	0	1	1	1	1
21	1	1	0	1	1	1	1
22	1	1	0	1	1	1	1

*NR significa que no respondió a la pregunta

Tabla C-7. Resultados obtenidos en el formulario sobre diagramas de clases de EC3-PREG

Participante	Caso						
	1	2	3	4	5	6	7
1	0	1	1	1	0	0	0
2	1	1	1	1	1	1	1
3	0	1	1	0	0	1	1
4	0	1	1	1	0	1	1
5	1	1	0	1	1	1	0
6	0	1	1	1	0	0	1
7	1	1	0	1	0	1	0
8	0	1	1	1	1	1	1
9	0	1	0	1	0	0	0
10	0	1	1	1	1	1	0
11	0	1	0	1	1	0	0
12	1	1	1	1	1	1	1
13	0	1	0	1	1	0	0
14	1	1	0	1	1	1	1
15	1	1	1	1	0	0	1
16	0	1	1	1	1	1	1
17	1	1	1	1	1	1	1
18	0	1	1	1	1	0	1
19	1	1	1	1	1	1	1
20	1	1	1	1	1	1	1
21	1	1	1	1	1	1	1
22	1	1	1	1	1	1	1

La Tabla C-8 muestra los resultados que se obtuvieron en la pregunta del cuestionario. Esta pregunta corresponde a seleccionar cuál de las dos técnicas es más clara de aplicar. La respuesta a) corresponde a E-R, la respuesta b) a diagramas de clases y c) que las dos son iguales.

Tabla C-8. Resultados obtenidos del cuestionario de EC3-PREG

Participante	Respuesta Pregunta 10
1	c
2	a
3	a
4	b
5	a
6	b

Participante	Respuesta Pregunta 10
7	a
8	b
9	a
10	b
11	b
12	a
13	a
14	a
15	b
16	c
17	b
18	b
19	a
20	b
21	No entregó cuestionario
22	a

Las imágenes escaneadas de los cuestionarios llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

C.4 Evaluación con Estudiantes de Posgrado: EC4-POST

Esta sección presenta la información complementaria del estudio realizado con estudiantes de posgrado EC4-POST. Dado que los formularios, cuestionario y casos fueron los mismos que los utilizados en el estudio EC3-PREG, solo se incluyen en esta sección las tablas con la información obtenida de los formularios y cuestionarios. Las imágenes escaneadas de los formularios y cuestionarios completados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

La Tabla C-9 y la Tabla C-10 muestran los resultados obtenidos en los formularios que completaron los participantes al experimento controlado. Un valor de “1” indica que identificó correctamente los ficheros y sus RET y “0” que no lo identificó de manera correcta.

Tabla C-9. Resultados obtenidos en el formulario E-R de EC4-POST

Participante	Caso						
	1	2	3	4	5	6	7
1	1	1	0	1	1	1	0
2	1	1	0	1	0	1	1
3	1	1	0	1	1	0	NR
4	1	1	0	1	1	0	0
5	1	1	0	1	0	0	1
6	1	0	1	1	1	1	0
7	0	1	0	1	0	0	0
8	1	1	1	1	1	1	0
9	0	1	0	1	0	1	0
10	0	0	0	0	0	1	1
11	0	0	0	0	0	1	1
12	1	1	0	1	0	0	0
13	0	0	0	0	0	1	1
14	0	1	0	1	1	0	0

*NR significa que no respondió a la pregunta

Tabla C-10. Resultados obtenidos en el formulario sobre diagramas de clases de EC4-POST

Participante	Caso						
	1	2	3	4	5	6	7
1	0	1	1	1	1	1	0
2	0	1	1	1	1	1	1
3	1	1	1	0	0	1	1
4	0	1	1	0	1	1	0
5	1	1	1	1	1	1	1
6	1	1	1	1	1	1	0
7	1	1	1	1	0	1	0
8	0	1	1	1	1	1	0
9	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1
11	0	1	1	1	1	1	1
12	0	0	0	1	1	1	1
13	1	0	0	0	0	1	0
14	0	1	0	1	1	0	0

La Tabla C-11 muestra los resultados que se obtuvieron en la pregunta del cuestionario. Esta pregunta corresponde a seleccionar cuál de las dos técnicas es más clara de aplicar. La respuesta a) corresponde a E-R, la respuesta b) a diagramas de clases y c) que las dos son iguales.

Tabla C-11. Resultados obtenidos del cuestionario de EC4-POST

Participante	Respuesta Pregunta 10
1	b
2	a
3	b
4	c
5	b
6	b
7	c
8	b
9	c
10	c
11	b
12	b
13	b
14	No completo

C.5 Evaluación con Estudiantes de Posgrado: EC5-POST

Esta sección presenta la información complementaria del estudio realizado con estudiantes de posgrado EC5-POST. Dado que los formularios empleados fueron los mismos que los utilizados en el estudio EC1-PROF, no se incluyen en esta sección.

Para este caso, se utilizaron los casos empleados en el estudio EC1-PROF. Según los resultados obtenidos en el estudio realizado con los docentes EP2-D0C, se pudo determinar que el primer caso correspondiente a diagramas de clases que se iba a utilizar tendía que ser modificado. La Figura C-22 muestra la primera página de los casos de diagramas de clases que fue modificada. No se incluyen las otras páginas ni los casos con diagramas E-R, porque estos no se modificaron (ver Subsección C.1.1 para ver el resto de páginas y los casos con diagramas E-R).

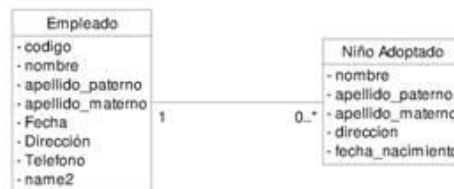
Las imágenes escaneadas de los cuestionarios llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

Casos para la Identificación de Ficheros en Puntos de Función para Diagrama de Clases de Análisis

Situación 1

Una compañía decide realizar un programa de "adopción" de niños, de manera que un trabajador pueda "adoptar" la cantidad de niños que quiera para darles juguetes en Navidad y en su cumpleaños. Un niño no puede ser adoptado por más de un trabajador y todos los niños siempre deberán tener un padre o madre adoptivo. Si a un empleado se le retira de la empresa, el o los niños que fueron adoptados por ese empleado son asignados a otra persona.

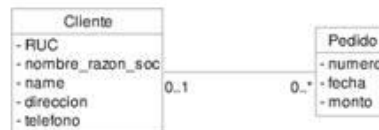
La información del empleado es la siguiente: código de trabajador, nombre del trabajador, apellido paterno, apellido materno, fecha, Dirección y teléfono. La información del niño adoptado es la siguiente: nombre, apellido paterno, apellido materno, dirección y fecha de nacimiento.



Situación 2

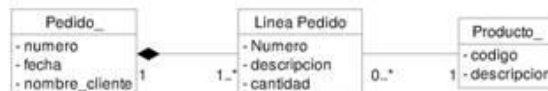
En un sistema de ventas, sólo se quiere automatizar lo siguiente: el registro de clientes y el registro de los datos principales de un pedido de cada uno de los clientes. La información del cliente es la siguiente: RUC, nombre o razón social, dirección y teléfono. La información del pedido es: número, fecha y monto total del pedido.

En este caso, pueden existir clientes sin pedidos y pedidos sin clientes.



Situación 3

Se quiere realizar un sistema que permita el registro de pedidos de los clientes. La información de los pedidos es: número, fecha y nombre del cliente. Adicionalmente cada línea del pedido (detalle del pedido) contendrá: número de línea, descripción del producto y cantidad del producto. El sistema también debe permitir el registro de los tipos de productos que vende, cuya información es: código y descripción.



José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DM-4051

Figura C-22. Caso con diagramas de clases para EC5-POST - Pág. 1/3

La Figura C-23 muestra el cuestionario que fue empleado para el estudio con EC5-POST. Este cuestionario permite evaluar en tres aspectos diferentes las técnicas utilizadas: facilidad de uso, utilidad e intención de uso.

Cuestionario sobre PF

Nombre: _____

	-				+
1. Suponiendo el caso que ya tenga listos un diagrama E-R y un diagrama de clases de análisis ¿La técnica orientada a objetos (técnica que usa emplea diagramas de clases de análisis) es más fácil de usar que la que emplea un diagrama E-R?	0	1	2	3	4
2. La técnica que emplea clases de análisis es más útil que la técnica E-R (Utilidad es el grado en el cual una técnica será efectiva para lograr los objetivos propuestos, en este caso determinar los puntos de función)	0	1	2	3	4
3. ¿Usaría la técnica que emplea clases para siguientes proyectos de desarrollo de software y no la emplea E-R?	0	1	2	3	4

Comentarios:

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú

1

Figura C-23. Cuestionario empleado para EC5-POST

La Tabla C-12 y la Tabla C-13 muestran los resultados obtenidos en los formularios que completaron los participantes al experimento controlado. Un valor de “1” indica que identificó correctamente los ficheros y sus RET y “0” que no lo identificó de manera correcta.

Tabla C-12. Resultados obtenidos en el formulario E-R de EC5-POST

Participante	Caso						
	1	2	3	4	5	6	7
1	1	1	1	1	1	1	1
2	0	1	1	0	0	0	0
3	1	1	1	1	1	0	0
4	1	1	0	1	0	0	0
5	1	1	0	1	0	1	1
6	0	1	1	1	0	0	0
7	1	1	0	1	1	1	1
8	1	0	0	0	0	1	0
9	1	1	1	1	1	0	1
10	1	1	1	1	0	1	0
11	1	1	0	1	1	1	0
12	1	1	0	1	0	1	0
13	1	1	1	1	1	1	1
14	1	1	1	1	1	1	0
15	1	1	1	1	1	1	1
16	0	0	0	0	0	0	0

Tabla C-13. Resultados obtenidos en el formulario sobre diagramas de clases de EC5-POST

Participante	Caso						
	1	2	3	4	5	6	7
1	0	1	1	1	0	1	1
2	0	1	0	1	1	1	0
3	1	1	1	1	1	1	0
4	1	1	0	1	1	1	0
5	0	1	0	1	0	1	1
6	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1
8	0	0	0	0	0	1	1
9	0	1	1	0	0	1	1
10	0	1	1	1	0	1	0
11	0	1	1	1	1	1	0
12	0	1	1	1	1	1	0

Participante	Caso						
	1	2	3	4	5	6	7
13	0	1	1	1	1	1	1
14	0	0	1	1	1	1	1
15	0	1	1	1	1	1	1
16	1	1	0	0	1	0	0

<p>Información Complementaria del Capítulo 6</p>	Anexo
	D

En este anexo se incluye información complementaria relacionada a la evaluación de UCPD (ver capítulo 6). La Sección D.1 contiene información relacionada al estudio realizado con alumnos de pregrado EI-PREG; la Sección C.1, información asociada al estudio realizado con profesionales E1-PROF; la Sección C.2, , información relacionada al estudio realizado con estudiantes de posgrado EC2-POST; y la Sección D.4, información relacionada al estudio con estudiantes de pregrado EC3-PREG.

D.1 Evaluación con Alumnos de Pregrado: EI-PREG

La Figura D-1 muestra el cuestionario que tuvieron que completar los alumnos de pregrado y en el que tenían que evaluar UCPD. La Tabla D-1 contiene el puntaje que asignó cada alumno a la pregunta 5 del cuestionario, para ello se trabajó con un rango del 1 al 5 para dicha calificación (escala de Likert). En la de la Figura D-2, se muestra como ejemplo el cuestionario llenado por uno de los alumnos de pregrado que participaron en el estudio. Las imágenes escaneadas de los cuestionarios llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

Cuestionario A

Ingeniería de Software

Nombre y apellidos: _____

Nota: No existe una respuesta "correcta" a estas preguntas, sólo responda con su opinión honesta sobre las preguntas realizadas

1. Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones

a) Maestros-transacciones-reportes b) Maestros-reportes-transacciones
c) Reportes-maestros-transacciones d) Reportes-transacciones-maestros
e) Transacciones-reportes-maestros f) Transacciones-maestros-reportes

2. ¿Ud. participó en la realización del diagrama de precedencias?

a) Sí b) No

Si en esta pregunta respondió "sí", pase a la pregunta 5

3. Responda a esta pregunta si es que en la anterior respondió "No" ¿Ud. revisó el diagrama de precedencias realizado en su equipo en algún momento del proyecto?

a) Sí b) No

4. Responda a esta pregunta si es que en la anterior respondió "Sí" ¿Entendió cuál era el objetivo del diagrama?

a) Sí b) No

5. Marque con una "X" el círculo que esté más cercano a su opinión.

a. Me parece difícil y complejo construir el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Me parece fácil y sencillo construir el diagrama de precedencias
b. Creo que el diagrama de precedencias NO es útil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Creo que el diagrama de precedencias es muy útil para planificar las iteraciones de construcción
c. En general, me parece útil el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	En general, me parece que el diagrama de precedencias NO es útil
d. Me hubiese parecido fácil construir los casos de uso sin seguir la secuencia sugerida por el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Me hubiese parecido difícil construir los casos de uso sin seguir la secuencia sugerida por el diagrama de precedencias
e. Usaré el diagrama de precedencias en próximos proyectos de desarrollo de software	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Definitivamente no usaré el diagrama de precedencias en próximos proyectos de desarrollo de software

7. ¿En qué casos cree que el diagrama de precedencias no sería útil?

6. ¿Cuáles son las razones por las que usaría o no usaría el diagrama de precedencias?

Figura D-1. Cuestionario empleado en EI-PREG

Tabla D-1. Calificación que dieron los alumnos por pregunta del cuestionario en EI-PREG

Observación	Resultados de la Pregunta 5				
	a	b	c	d	e
1	4	5	5	3	4
2	3	5	5	4	4
3	3	5	5	3	4
4	4	4	4	3	4
5	5	5	5	4	5
6	4	3	3	3	5
7	4	5	5	5	5
8	3	5	5	4	5
9	5	5	5	3	5
10	5	5	5	3	4
11	3	4	4	4	4
12	3	5	5	4	4
13	3	5	5	3	4
14	5	5	5	5	5
15	4	4	4	3	4
16	5	5	5	4	5
17	4	5	4	4	5
18	4	4	4	3	4
19	3	4	4	3	3
20	4	5	3	1	5
21	5	3	3	3	3
22	4	5	2	3	4
23	4	5	5	4	5
24	5	5	5	4	5
25	4	4	4	3	4
26	3	4	4	3	5
27	4	5	3	3	2
28	5	4	5	4	4
29	3	2	3	3	3
30	4	5	5	4	5
31	4	3	3	3	3

Cuestionario A
Ingeniería de Software

Nombre y apellidos: Marco Araujo Parra

Nota: No existe una respuesta "correcta" a estas preguntas, sólo responda con su opinión honesta sobre las preguntas realizadas

1. Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones

☒ a) Maestros-transacciones-reportes
☐ b) Maestros-reportes-transacciones
☐ c) Reportes-maestros-transacciones
☐ d) Reportes-transacciones-maestros
☐ e) Transacciones-reportes-maestros
☐ f) Transacciones-maestros-reportes

2. ¿Ud. participó en la realización del diagrama de precedencias?
a) Si ☒ No

Si en esta pregunta respondió "sí", pase a la pregunta 5

3. Responda a esta pregunta si es que en la anterior respondió "No" ¿Ud. revisó el diagrama de precedencias realizado en su equipo en algún momento del proyecto?
☒ a) Si ☐ b) No

4. Responda a esta pregunta si es que en la anterior respondió "Sí" ¿Entendió cuál era el objetivo del diagrama?
a) Si ☐ b) No

5. Marque con una "X" el círculo que esté más cercano a su opinión.

a. Me parece difícil y complejo construir el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Me parece fácil y sencillo construir el diagrama de precedencias
b. Creo que el diagrama de precedencias NO es útil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Creo que el diagrama de precedencias es muy útil para planificar las iteraciones de construcción
c. En general, me parece útil el diagrama de precedencias	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	En general, me parece que el diagrama de precedencias NO es útil
d. Me hubiese parecido fácil construir los casos de uso sin seguir la secuencia sugerida por el diagrama de precedencias	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Me hubiese parecido difícil construir los casos de uso sin seguir la secuencia sugerida por el diagrama de precedencias
e. Usaré el diagrama de precedencias en próximos proyectos de desarrollo de software	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Definitivamente no usaré el diagrama de precedencias en próximos proyectos de desarrollo de software

7. ¿En qué casos cree que el diagrama de precedencias no sería útil?
En casos en los que el área posea pocas entidades y no se puedan identificar niveles entre ellas.

6. ¿Cuáles son las razones por las que usaría o no usaría el diagrama de precedencias?

Figura D-2. Cuestionario del participante 1 de EI-PREG

D.2 Evaluación con Profesionales: EC1-PROF

En esta sección se presenta la información adicional del estudio realizado con profesionales EC1-PROF. Inicialmente se muestran los casos de estudio que

fueron empleados por los profesionales, luego los cuestionarios que tuvieron que completar y, finalmente, los cuestionarios que fueron llenados por un participante de la evaluación, el cual se incluye como ejemplo. También, se incluyen los diagramas que realizó este participante de la evaluación para poder responder a los cuestionarios.

Las imágenes escaneadas de los cuestionarios llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

D.2.1 Casos de estudio

Las figuras que se muestran a continuación presentan los casos de estudio que fueron entregados a los participantes del estudio. Las tres primeras figuras (Figura D-3, Figura D-4 y Figura D-5) muestran las páginas del primer caso de estudio que corresponde a un sistema de ventas para un restaurante. Las tres figuras últimas de esta sección (Figura D-6, Figura D-7 y Figura D-8) presentan las páginas del segundo caso de estudio, el cual corresponde a un sistema de información de matrícula para un colegio de educación secundaria.

Caso de estudio Sistema de información para un restaurante

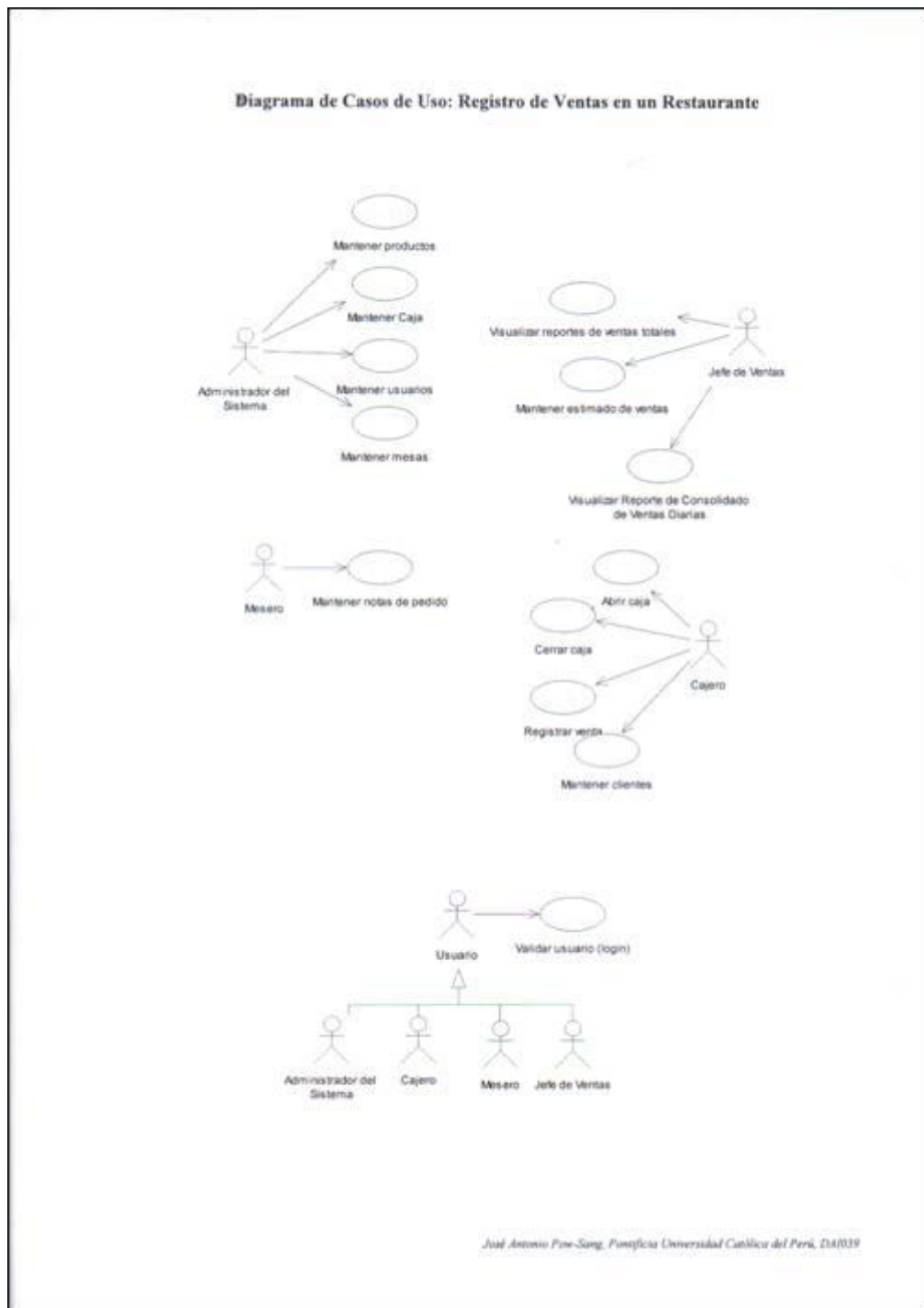


Figura D-3. Caso de estudio sistema de restaurante- Pág. 1/3

<p>1. Caso de Uso Registrar Venta Este caso de uso tiene por objetivo registrar ventas. Este caso de uso es iniciado por el cajero.</p> <p>Precondiciones</p> <ul style="list-style-type: none"> • Se ha debido validar el usuario con perfil de cajero • Se ha debido abrir la caja <p>Información asociada al caso de uso: boleta, factura, cliente, producto, usuario</p>
<p>2. Caso de Uso Generar Reporte de Ventas Totales Este caso de uso tiene por objetivo generar un reporte de las ventas totales configurable por un periodo de tiempo. Este caso de uso es iniciado por el jefe de ventas.</p> <p>Precondiciones</p> <ul style="list-style-type: none"> • Se ha debido validar el usuario <p>Información asociada al caso de uso: boleta, factura, producto, usuario</p>
<p>3. Caso de Uso Visualizar Consolidado de Ventas Este caso de uso tiene por objetivo ver el consolidado de ventas de los productos vendidos en un día para la sucursal del usuario. Este caso de uso es iniciado por el administrador.</p> <p>Precondiciones</p> <ul style="list-style-type: none"> • Se ha debido validar el usuario. <p>Información asociada al caso de uso: boleta, factura, producto, usuario</p>
<p>4. Caso de uso Abrir Caja Este caso de uso tiene por objetivo registrar el monto inicial de una caja al inicio de un turno en el sistema. Este caso de uso es iniciado por el cajero.</p> <p>Precondiciones</p> <ul style="list-style-type: none"> • Se ha debido validar el usuario <p>Información asociada al caso de uso: caja, usuario</p>
<p>5. Caso de uso Cerrar Caja Este caso de uso tiene por objetivo registrar el monto de efectivo de una caja al finalizar un turno en el sistema. Este caso de uso es iniciado por el cajero.</p> <p>Precondiciones</p> <ul style="list-style-type: none"> • Se ha debido validar el usuario • Se ha debido abrir la caja <p>Información asociada al caso de uso: caja, factura, boleta, usuario</p>
<p>6. Caso de Uso Mantener Cajas Este caso de uso tiene por objetivo registrar nuevas cajas así como también modificar y eliminar las mismas. Este caso de uso es iniciado por el administrador.</p> <p>Precondiciones</p> <ul style="list-style-type: none"> • Se ha debido validar el usuario <p>Información asociada al caso de uso: caja, usuario</p>
<p>7. Caso de Uso Mantener Clientes Este caso de uso tiene por objetivo registrar nuevos clientes así como también la modificación y eliminación de la información de los mismos. El caso de uso es iniciado por el cajero.</p>

José Antonio Pow Sang, Pontificia Universidad Católica del Perú, DA0019

Figura D-4. Caso de estudio sistema de restaurante - Pág. 2/3

<p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: cliente, usuario</p>
<p>8. Caso de Uso Mantener Productos Este caso de uso tiene por objetivo registrar, modificar o eliminar productos (platos, bebidas, combos, etc). El caso de uso es iniciado por el Jefe de Ventas.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: producto, usuario</p>
<p>9. Caso de Uso Mantener Nota de Pedido Este caso de uso tiene por objetivo registrar nuevas notas de pedido así como también la modificación o eliminación de la información de las notas de pedido ya existentes, mientras aún no se haya generado un documento de pago. El caso de uso es iniciado por el mesero y cajero.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: nota de pedido, mesa, usuario</p>
<p>10. Caso de Uso Mantener Estimado de Ventas Este caso de uso tiene por objetivo registrar las cantidades de productos estimadas a vender para una fecha o rangos de fecha determinados, también permite modificar y eliminar las mismas. Este caso de uso es iniciado por el administrador.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: estimado de venta, boleta, factura, producto, usuario</p>
<p>11. Caso de Uso Mantener Mesa Este caso de uso tiene por objetivo registrar nuevas mesas así como también modificar y eliminar las mismas. El caso de uso es iniciado por el Jefe de Ventas.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: mesa, usuario</p>
<p>12. Caso de Uso Validar Usuario Este caso de uso corresponde a la validación de usuarios (login del sistema).</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se han debido registrar los usuarios <p>Información asociada al caso de uso: usuario.</p>
<p>13. Caso de Uso Mantener Usuarios Este caso de uso corresponde al registro y eliminación de usuarios del sistema.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario como administrador. <p>Información asociada al caso de uso: usuario.</p>

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, D10039

Figura D-5. Caso de estudio sistema de restaurante - Pág. 3/3

Caso de estudio Sistema de matrícula para un colegio

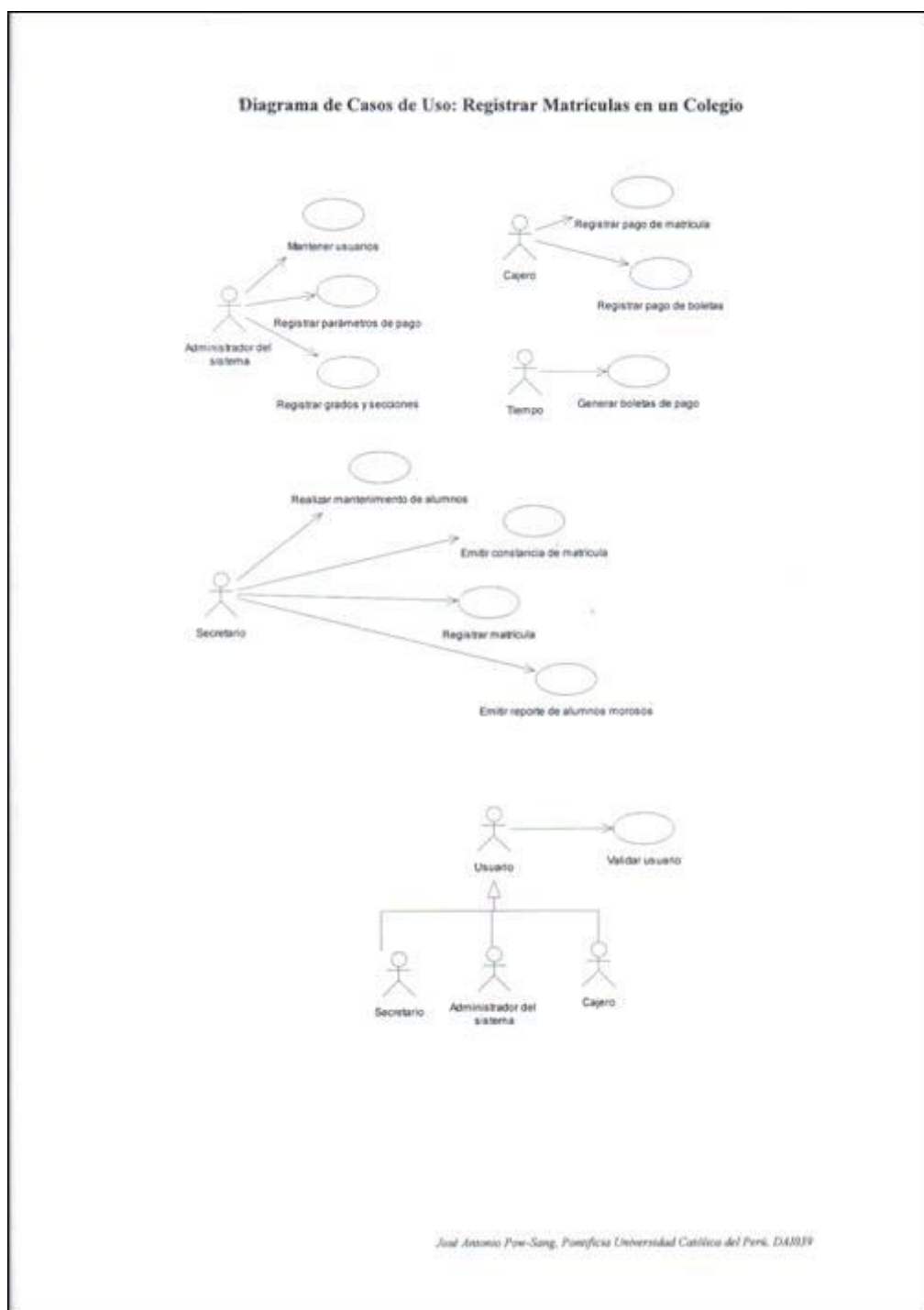


Figura D-6. Caso de estudio sistema de matrícula para un colegio - Pág. 1/3

<p>1. Caso de Uso Validar Usuario Este caso de uso corresponde a la validación de usuarios (login del sistema).</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se han debido registrar los usuarios. <p>Información asociada al caso de uso: usuario.</p>
<p>2. Caso de Uso Mantener Usuarios Este caso de uso corresponde al registro y eliminación de usuarios del sistema.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario como administrador. <p>Información asociada al caso de uso: usuario.</p>
<p>3. Registrar grados y secciones. El propósito de este caso de uso es el de registrar en el sistema los grados y secciones con las que trabajará el colegio en un año en particular.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: grado, sección.</p>
<p>4. Registrar parámetros de pagos. El propósito de este caso de uso es el de establecer los parámetros de pagos que manejará el colegio durante un año determinado; como la forma de generación de las boletas de pago (manual o automática), periodo de generación de boletas, los conceptos de pago y el tipo de cambio de soles a dólares, las escalas de pago que manejará el colegio y los tipos de beneficios (descuentos) que se le podría otorgar al estudiante (los beneficios se aplican a determinados alumnos, por ejemplo a deportistas).</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: parámetro.</p>
<p>5. Registrar matrícula. El propósito de este caso de uso es matricular a un alumno y generar la boleta. Si es alumno antiguo se procederá a la ratificación de su matrícula en caso contrario se ingresará la información del alumno nuevo al sistema.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: matrícula, alumno.</p>
<p>6. Registrar pago de matrícula. El propósito de este caso de uso es registrar el pago de la matrícula con lo cual se confirma la matrícula y se asigna la vacante al alumno.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: matrícula, pago de matrícula, alumno.</p>
<p>7. Emitir constancia de matrícula. El propósito de este caso de uso es emitir la constancia de matrícula con la información detallada de un determinado alumno.</p> <p>Precondiciones</p> <ul style="list-style-type: none">• Se ha debido validar el usuario <p>Información asociada al caso de uso: matrícula, alumno.</p>

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DA039

Figura D-7. Caso de estudio sistema de matrícula para un colegio - Pág. 2/3

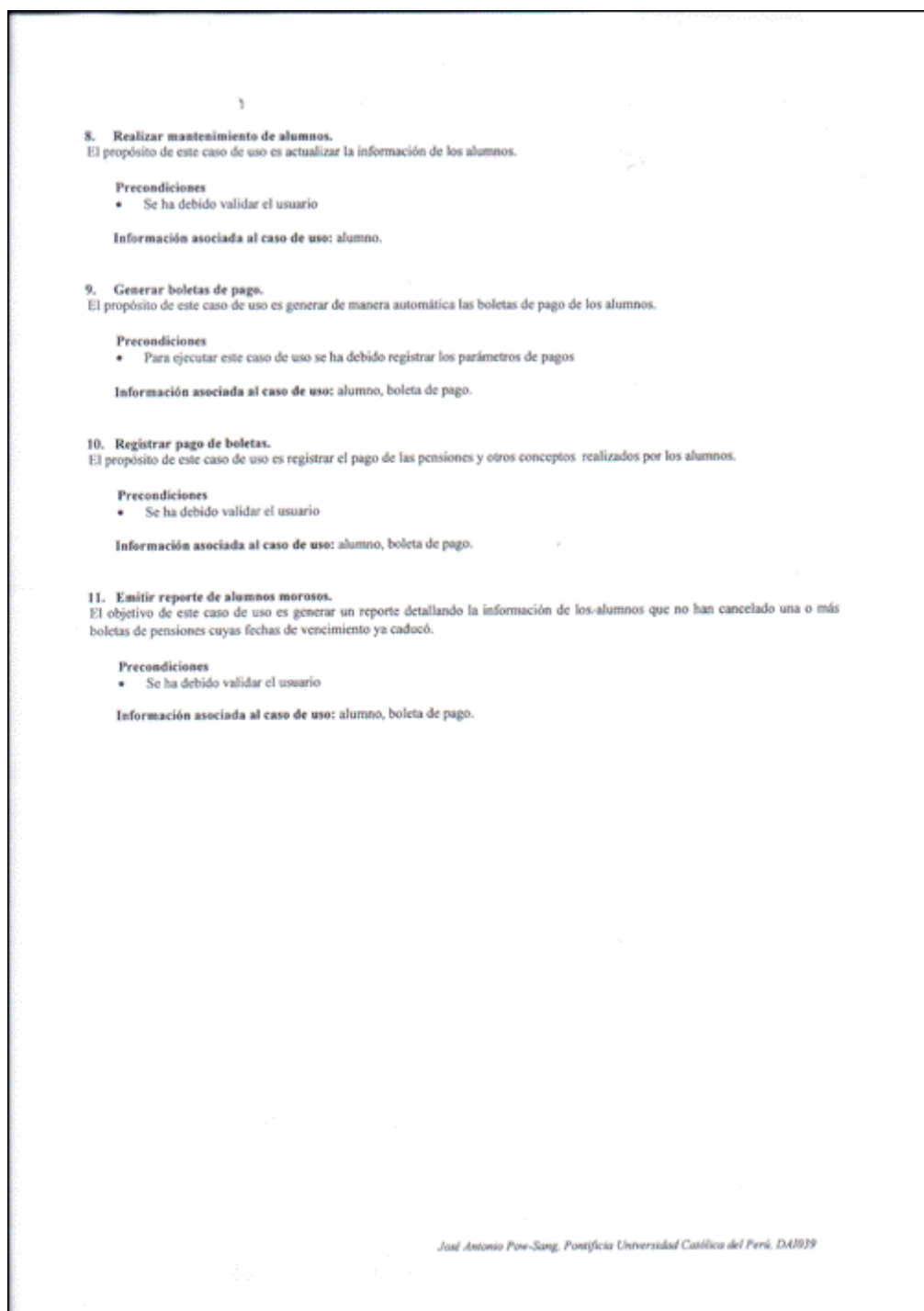


Figura D-8. Caso de estudio sistema de matrícula para un colegio - Pág. 3/3

D.2.2 Cuestionarios

Las figuras que se muestran a continuación (Figura D-9, Figura D-10, Figura D-11 y Figura D-12) presentan los cuestionarios que tuvieron que llenar los participantes del estudio.

Cuestionario 1

Nombre y apellidos: _____

De acuerdo a su experiencia y considerando que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

1. ¿Construiría primero "cerrar caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
2. ¿Construiría primero "mantener caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
3. ¿Construiría primero "mantener notas de pedido" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
4. ¿Construiría primero "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
5. ¿Construiría primero "registrar venta" antes de "cerrar caja"?
a) Sí b) No c) Me es indiferente
6. ¿Construiría primero "mantener mesas" antes de "mantener notas de pedido"?
a) Sí b) No c) Me es indiferente
7. ¿Realizaría las pruebas de "cerrar caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
8. ¿Realizaría las pruebas de "mantener caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
9. ¿Realizaría las pruebas de "mantener notas de pedido" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
10. ¿Realizaría las pruebas de "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
11. ¿Realizaría las pruebas de "registrar venta" antes de "cerrar caja"?
a) Sí b) No c) Me es indiferente
12. ¿Realizaría las pruebas de "mantener mesas" antes de "mantener notas de pedido"?
a) Sí b) No c) Me es indiferente

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DAI039

Figura D-9. Cuestionario 1 para EC1-PROF

1

Cuestionario 2

Nombre y apellidos: _____

De acuerdo a su experiencia al diagrama de precedencias y considerando que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

1. ¿Construiría primero "registrar matrícula" antes de "registrar grados y secciones"?
a) Sí b) No c) Me es indiferente
2. ¿Construiría primero "registrar pago de matrícula" antes de "registrar matrícula"?
a) Sí b) No c) Me es indiferente
3. ¿Construiría primero "realizar mantenimiento de alumnos" antes de "registrar matrícula"?
a) Sí b) No c) Me es indiferente
4. ¿Construiría primero "emitir reporte de alumnos morosos" antes de "realizar mantenimiento de alumnos"?
a) Sí b) No c) Me es indiferente
5. ¿Construiría primero "emitir constancia de matrícula" antes de "registrar pago de matrícula"?
a) Sí b) No c) Me es indiferente
6. ¿Realizaría las pruebas de "registrar matrícula" antes de "registrar grados y secciones"?
a) Sí b) No c) Me es indiferente
7. ¿Realizaría las pruebas de "registrar pago de matrícula" antes de "registrar matrícula"?
a) Sí b) No c) Me es indiferente
8. ¿Realizaría las pruebas de "realizar mantenimiento de alumnos" antes de "registrar matrícula"?
a) Sí b) No c) Me es indiferente
9. ¿Realizaría las pruebas de "emitir reporte de alumnos morosos" antes de "realizar mantenimiento de alumnos"?
a) Sí b) No c) Me es indiferente
10. ¿Realizaría las pruebas de "emitir constancia de matrícula" antes de "registrar pago de matrícula"?
a) Sí b) No c) Me es indiferente

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DABSP 1

Figura D-10. Cuestionario 2 para EC1-PROF

1

Cuestionario 3

Nombre y apellidos: _____

De acuerdo a su experiencia al diagrama de precedencias y considerando que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

1. ¿Construiría primero "cerrar caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
2. ¿Construiría primero "mantener caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
3. ¿Construiría primero "registrar venta" antes de "mantener clientes"?
a) Sí b) No c) Me es indiferente
4. ¿Construiría primero "mantener notas de pedido" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
5. ¿Construiría primero "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
6. ¿Construiría primero "registrar venta" antes de "cerrar caja"?
a) Sí b) No c) Me es indiferente
7. ¿Construiría primero "mantener mesas" antes de "mantener notas de pedido"?
a) Sí b) No c) Me es indiferente
8. ¿Realizaría las pruebas de "cerrar caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
9. ¿Realizaría las pruebas de "mantener caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
10. ¿Realizaría las pruebas de "registrar venta" antes de "mantener clientes"?
a) Sí b) No c) Me es indiferente
11. ¿Realizaría las pruebas de "mantener notas de pedido" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
12. ¿Realizaría las pruebas de "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
13. ¿Realizaría las pruebas de "registrar venta" antes de "cerrar caja"?
a) Sí b) No c) Me es indiferente
14. ¿Realizaría las pruebas de "mantener mesas" antes de "mantener notas de pedido"?
a) Sí b) No c) Me es indiferente

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DA039

Figura D-11. Cuestionario 3 para EC1-PROF

1

Cuestionario 4

Nombre y apellidos: _____

Indique cuánto tiempo de experiencia en años tiene en desarrollo de software: _____

Indique si trabaja con casos de uso: _____

CONSTRUCCIÓN DE CASOS DE USO

1. Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones

a) Maestros-transacciones-reportes b) Maestros-reportes-transacciones c) Reportes-maestros-transacciones

d) Reportes-transacciones-maestros e) Transacciones-reportes-maestros f) Transacciones-maestros-reportes

2. Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones

a) Maestros-transacciones-reportes b) Maestros-reportes-transacciones c) Reportes-maestros-transacciones

d) Reportes-transacciones-maestros e) Transacciones-reportes-maestros f) Transacciones-maestros-reportes

DIAGRAMA DE PRECEDENCIAS

	-				+
3. Facilidad para construir o realizar el diagrama de precedencias	0	1	2	3	4
4. Utilidad del diagrama de precedencias para determinar la secuencia de construcción de software	0	1	2	3	4
5. Utilidad del diagrama de precedencias para determinar la secuencia de pruebas	0	1	2	3	4
6. Para los proyectos que haya realizado, ¿le hubiese parecido muy complicado el construir el software sin seguir la secuencia sugerida por el diagrama de precedencias? (0=nada complicado, 4=muy complicado)	0	1	2	3	4
7. ¿Usaría el diagrama de precedencias para siguientes proyectos de desarrollo de software?	0	1	2	3	4

COMENTARIOS Y SUGERENCIAS DEL DIAGRAMA DE PRECEDENCIAS

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DA0039

Figura D-12. Cuestionario 4 para EC1-PROF

En el cuestionario mostrado en la figura anterior, Figura, se puede observar que las preguntas 1 y 2 son las mismas. Este fue un error que se detectó durante la ejecución del experimento y fue comentado a los participantes.

D.2.3 Tablas con información obtenida de los cuestionarios

La Tabla D-2, la Tabla D-3, la Tabla D-4 y la Tabla D-5 muestran los resultados obtenidos en los tres primeros cuestionarios del estudio. Un valor de “1” indica que acertó en la respuesta y “0” es respuesta errónea. Cabe resaltar que no se han incluido los resultados de los cuestionarios en que los participantes contestaron una secuencia diferente a *maestros-transacciones-reportes*.

Aunque en los cuestionarios 1, 2 y 3 incluyeron preguntas relacionadas a la secuencia de realización de las pruebas, solo se han empleado las respuestas de las preguntas sobre construcción. Los resultados obtenidos sobre la parte de pruebas se emplearán en trabajos futuros.

Tabla D-2. Resultados obtenidos en el cuestionario 1 de EC1-PROF

Participante	Pregunta						Porcentaje de Respuestas correctas
	1	2	3	4	5	6	
1	0	0	0	0	0	0	0.0%
2	1	0	0	1	0	0	33.3%
3	1	1	1	1	1	1	100.0%
4	1	1	1	1	1	1	100.0%
5	1	1	1	0	1	1	83.3%
6	1	0	1	1	1	1	83.3%
7	1	0	1	1	1	1	83.3%
8	1	0	1	1	1	1	83.3%
9	1	1	0	1	1	1	83.3%
10	1	0	1	1	1	1	83.3%
11	1	0	1	1	1	0	66.7%
12	1	1	0	1	0	0	50.0%
13	1	1	1	1	1	1	100.0%
14	1	0	1	0	1	1	66.7%
15	1	1	1	1	1	1	100.0%
16	1	1	1	1	0	1	83.3%
17	1	1	1	1	1	1	100.0%
18	1	1	1	1	1	0	83.3%
19	1	0	1	1	1	1	83.3%
20	1	0	1	1	0	1	66.7%
21	1	1	1	1	1	0	83.3%
22	1	1	1	1	1	1	100.0%
23	1	0	1	1	1	1	83.3%
24	1	0	1	1	1	0	66.7%
25	1	1	1	1	0	0	66.7%

Tabla D-3. Resultados obtenidos en el cuestionario 2 de EC1-PROF

Participante	Pregunta					Porcentaje de Respuestas correctas
	1	2	3	4	5	
1	1	1	1	0	1	80.0%
2	1	1	0	1	1	80.0%
3	1	1	0	0	1	60.0%
4	0	1	0	0	1	40.0%
5	1	1	1	1	1	100.0%
6	1	1	1	1	1	100.0%
7	0	1	1	0	1	60.0%
8	1	1	1	1	1	100.0%
9	1	1	1	1	1	100.0%
10	1	1	1	1	1	100.0%
11	1	1	1	1	1	100.0%
12	1	1	1	1	1	100.0%
13	1	1	1	1	1	100.0%
14	1	1	1	1	1	100.0%
15	1	1	1	1	1	100.0%
16	1	1	1	1	1	100.0%
17	1	1	1	1	1	100.0%
18	1	1	1	1	1	100.0%
19	1	1	0	0	1	60.0%
20	1	1	1	1	1	100.0%
21	1	1	1	1	1	100.0%
22	1	1	1	1	1	100.0%
23	1	1	1	1	1	100.0%
24	1	1	1	1	1	100.0%
25	1	0	1	1	1	80.0%

Tabla D-4. Resultados obtenidos en el cuestionario 3 de EC1-PROF

Participante	Pregunta						Porcentaje de Respuestas correctas
	1	2	3	4	5	6	
1	1	1	1	0	1	1	83.3%
2	1	1	1	1	1	1	100.0%
3	1	1	1	1	1	1	100.0%
4	1	1	1	1	0	1	83.3%
5	1	1	0	1	1	1	83.3%
6	1	0	1	1	1	0	66.7%
7	1	0	1	1	1	1	83.3%
8	1	1	0	1	1	1	83.3%
9	1	1	1	1	1	1	100.0%
10	1	1	1	1	0	1	83.3%
11	1	1	1	1	1	0	83.3%
12	1	1	1	1	1	1	100.0%
13	1	1	1	1	1	1	100.0%
14	1	1	1	1	0	1	83.3%
15	1	1	0	1	1	1	83.3%
16	1	1	1	1	1	1	100.0%
17	1	1	1	1	1	1	100.0%
18	1	1	1	1	0	1	83.3%
19	1	1	1	1	1	1	100.0%
20	1	1	1	1	0	1	83.3%
21	1	1	1	1	0	1	83.3%
22	1	1	0	1	1	1	83.3%
23	1	1	0	1	1	1	83.3%
24	1	1	1	1	0	1	83.3%
25	1	1	1	1	0	0	66.7%

Tabla D-5. Resultados obtenidos en el cuestionario 4 de EC1-PROF

Observación	Pregunta			
	2	3	5	6
1	3	4	1	3
2	2	3	4	4
3	3	3	0	4
4	2	4	3	3
5	3	3	4	4
6	2	3	NR*	3
7	2	2	2	2
8	3	4	2	4
9	3	3	2	3
10	3	3	3	3
11	3	4	4	4
12	3	4	3	3
13	3	4	3	4
14	4	3	3	4
15	3	3	2	3
16	3	4	2	4
17	4	4	2	3
18	4	3	0	3
19	3	3	2	4
20	3	4	2	3
21	4	4	0	4
22	2	2	2	2
23	4	4	2	3
24	1	3	3	0
25	1	3	2	3

*NR significa que no respondió a la pregunta

En la Tabla D-5, la pregunta 5 fue una pregunta que se presentó en sentido inverso a las otras preguntas y los valores que se muestran corresponde a valores que los profesionales seleccionaron. Para el procesamiento estadístico se tuvo que invertir los valores obtenidos en esa pregunta. Además, para el procesamiento estadístico, se sumó un “1” a todos los valores de la tabla anterior, a fin de poder comparar los resultados que fueron obtenidos con los alumnos de pregrado.

D.2.4 Cuestionarios llenados y diagramas realizados por los participantes

En la Figura D-13, la Figura D-14, la Figura D-15 y la Figura D-16, se muestran como ejemplo el cuestionario llenado por uno de los profesionales que participaron en el estudio. Los diagramas realizados por el participante se muestran en la Figura D-17 y la Figura D-18. Las imágenes escaneadas de todos

los participantes se encuentran en el siguiente enlace
<http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

Cabe resaltar que algunos diagramas no siguieron exactamente la notación explicada, debido a que se les comentó que no era necesario (por ejemplo, usar flechas continuas en vez de punteadas o no incluir “<<antecede>>”). Esto se hizo con el objetivo que pudieran responder a los cuestionarios de manera rápida y no detenerse con los detalles de la notación de UCPD.

Cuestionario 1

Nombre y apellidos: Alexander Ponilla Dionisio

De acuerdo a su experiencia y considerando que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

- ¿Construiría primero "cerrar caja" antes de "abrir caja"?
a) Sí b) No ☒ c) Me es indiferente
- ¿Construiría primero "mantener caja" antes de "abrir caja"?
a) Sí b) No ☒ c) Me es indiferente
- ¿Construiría primero "mantener notas de pedido" antes de "registrar venta"?
a) Sí b) No ☒ c) Me es indiferente
- ¿Construiría primero "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí b) No ☒ c) Me es indiferente
- ¿Construiría primero "registrar venta" antes de "cerrar caja"?
a) Sí b) No ☒ c) Me es indiferente
- ¿Construiría primero "mantener mesas" antes de "mantener notas de pedido"?
a) Sí b) No ☒ c) Me es indiferente
- ¿Realizaría las pruebas de "cerrar caja" antes de "abrir caja"?
a) Sí ☒ b) No c) Me es indiferente
- ¿Realizaría las pruebas de "mantener caja" antes de "abrir caja"?
☒ a) Sí b) No c) Me es indiferente
- ¿Realizaría las pruebas de "mantener notas de pedido" antes de "registrar venta"?
☒ a) Sí b) No c) Me es indiferente
- ¿Realizaría las pruebas de "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí ☒ b) No c) Me es indiferente
- ¿Realizaría las pruebas de "registrar venta" antes de "cerrar caja"?
☒ a) Sí b) No c) Me es indiferente
- ¿Realizaría las pruebas de "mantener mesas" antes de "mantener notas de pedido"?
☒ a) Sí b) No c) Me es indiferente

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DA019

Figura D-13. Cuestionario 1 del participante 1 de EC1-PROF

Cuestionario 2

Nombre y apellidos: Alexander Ponilla Dionisio

De acuerdo a su experiencia y considerando que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

- ¿Construiría primero "registrar matrícula" antes de "registrar grados y secciones"?
a) Si ☒ b) No ☐ c) Me es indiferente
- ¿Construiría primero "registrar pago de matrícula" antes de "registrar matrícula"?
a) Si ☒ b) No ☐ c) Me es indiferente
- ¿Construiría primero "realizar mantenimiento de alumnos" antes de "registrar matrícula"?
☒ a) Si ☐ b) No ☐ c) Me es indiferente
- ¿Construiría primero "emitir reporte de alumnos morosos" antes de "realizar mantenimiento de alumnos"?
a) Si ☐ b) No ☒ c) Me es indiferente
- ¿Construiría primero "emitir constancia de matrícula" antes de "registrar pago de matrícula"?
☒ a) Si ☐ b) No ☐ c) Me es indiferente
- ¿Realizaría las pruebas de "registrar matrícula" antes de "registrar grados y secciones"?
a) Si ☒ b) No ☐ c) Me es indiferente
- ¿Realizaría las pruebas de "registrar pago de matrícula" antes de "registrar matrícula"?
a) Si ☒ b) No ☐ c) Me es indiferente
- ¿Realizaría las pruebas de "realizar mantenimiento de alumnos" antes de "registrar matrícula"?
☒ a) Si ☐ b) No ☐ c) Me es indiferente
- ¿Realizaría las pruebas de "emitir reporte de alumnos morosos" antes de "realizar mantenimiento de alumnos"?
a) Si ☒ b) No ☐ c) Me es indiferente
- ¿Realizaría las pruebas de "emitir constancia de matrícula" antes de "registrar pago de matrícula"?
☒ a) Si ☐ b) No ☐ c) Me es indiferente

José Antonio Pow Sang, Pontificia Universidad Católica del Perú, D.A.R.O.39

Figura D-14. Cuestionario 2 del participante 1 de EC1-PROF

Cuestionario 3

Nombre y apellidos: Alexander Bonilla Dionisio

De acuerdo a su experiencia y considerando que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

1. ¿Construiría primero "cerrar caja" antes de "abrir caja"?
a) Sí ☐ b) No ☒ c) Me es indiferente ☐
2. ¿Construiría primero "mantener caja" antes de "abrir caja"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
3. ¿Construiría primero "registrar venta" antes de "mantener clientes"?
a) Sí ☐ b) No ☒ c) Me es indiferente ☐
4. ¿Construiría primero "mantener notas de pedido" antes de "registrar venta"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
5. ¿Construiría primero "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
6. ¿Construiría primero "registrar venta" antes de "cerrar caja"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
7. ¿Construiría primero "mantener mesas" antes de "mantener notas de pedido"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
8. ¿Realizaría las pruebas de "cerrar caja" antes de "abrir caja"?
a) Sí ☐ b) No ☒ c) Me es indiferente ☐
9. ¿Realizaría las pruebas de "mantener caja" antes de "abrir caja"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
10. ¿Realizaría las pruebas de "registrar venta" antes de "mantener clientes"?
a) Sí ☐ b) No ☐ c) Me es indiferente ☒
11. ¿Realizaría las pruebas de "mantener notas de pedido" antes de "registrar venta"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
12. ¿Realizaría las pruebas de "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí ☐ b) No ☐ c) Me es indiferente ☒
13. ¿Realizaría las pruebas de "registrar venta" antes de "cerrar caja"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐
14. ¿Realizaría las pruebas de "mantener mesas" antes de "mantener notas de pedido"?
a) Sí ☒ b) No ☐ c) Me es indiferente ☐

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DA009

Figura D-15. Cuestionario 3 del participante 1 de EC1-PROF

Cuestionario 4

Nombre y apellidos: Alexander Ponillo Dionisio

Indique cuánto tiempo de experiencia en años tiene en desarrollo de software: 9

Indique si trabaja con casos de uso: NO

CONSTRUCCIÓN DE CASOS DE USO

1. Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones

a) Maestros-transacciones-reportes b) Maestros-reportes-transacciones c) Reportes-maestros-transacciones
d) Reportes-transacciones-maestros e) Transacciones-reportes-maestros f) Transacciones-maestros-reportes

2. Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones

a) Maestros-transacciones-reportes b) Maestros-reportes-transacciones c) Reportes-maestros-transacciones
d) Reportes-transacciones-maestros e) Transacciones-reportes-maestros f) Transacciones-maestros-reportes

DIAGRAMA DE PRECEDENCIAS

	-				+
3. Facilidad para construir o realizar el diagrama de precedencias	0	1	2	3	4
4. Utilidad del diagrama de precedencias para determinar la secuencia de construcción de software	0	1	2	3	4
5. Utilidad del diagrama de precedencias para determinar la secuencia de pruebas	0	1	2	3	4
6. Para los proyectos que haya realizado, ¿le hubiese parecido muy complicado el construir el software sin seguir la secuencia sugerida por el diagrama de precedencias? (0=nada complicado, 4=muy complicado)	0	1	2	3	4
7. ¿Usaría el diagrama de precedencias para siguientes proyectos de desarrollo de software?	0	1	2	3	4

COMENTARIOS Y SUGERENCIAS DEL DIAGRAMA DE PRECEDENCIAS

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, DA0019

Figura D-16. Cuestionario 4 del participante 1 de EC1-PROF

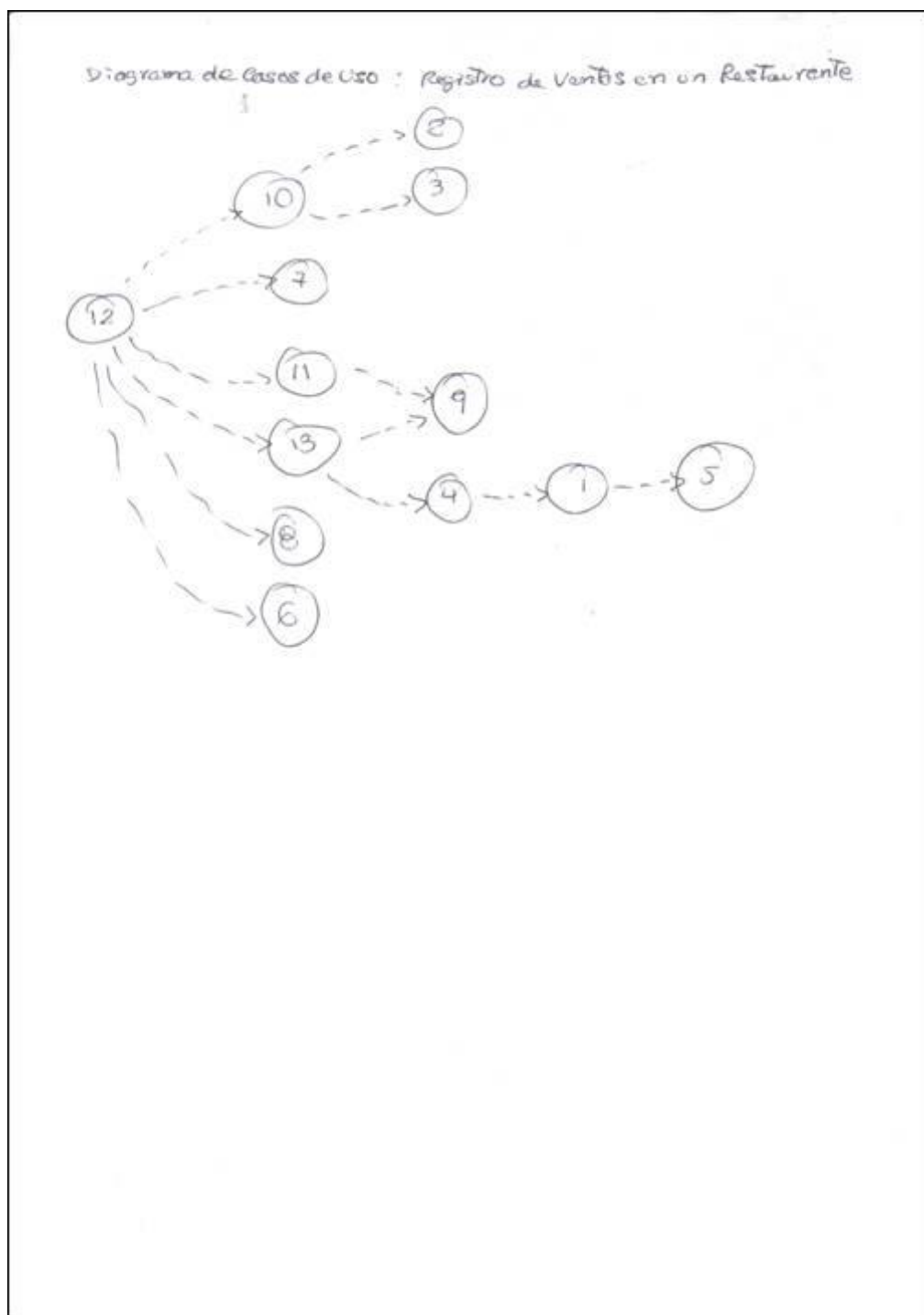


Figura D-17. UCPD para el primer caso de estudio del participante 1 de EC1-PROF

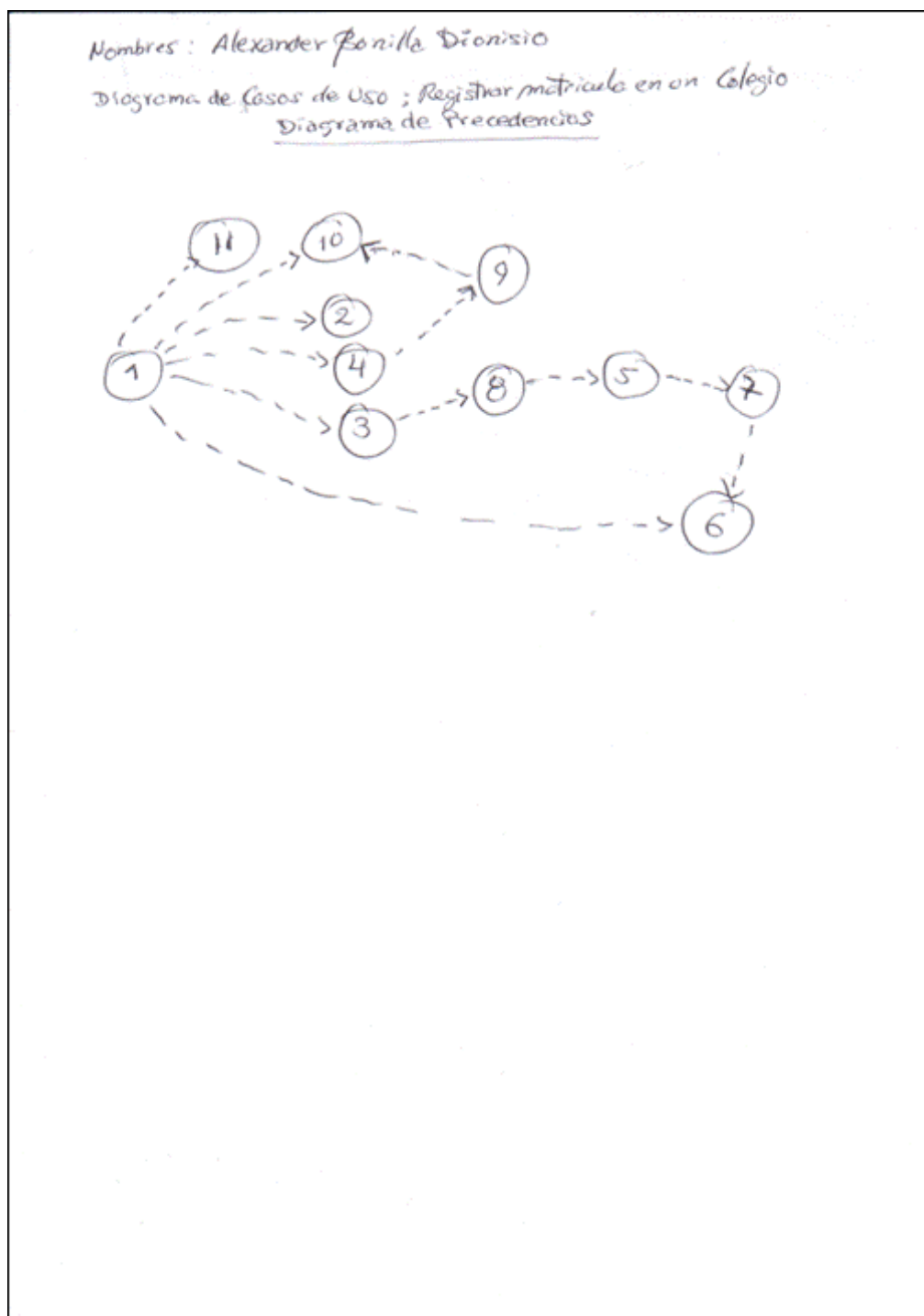


Figura D-18. UCPD para el segundo caso de estudio del participante 1 de EC1-PROF

D.3 Evaluación con Estudiantes de Posgrado: EC2-POST

En esta sección se presenta la información adicional del estudio realizado con estudiantes de posgrado EC2-POST. Se incluyen los cuestionarios que tuvieron que completar los participantes y las tablas con la información que se obtuvo de los cuestionarios. Dado que los casos de estudio fueron los mismos que los empleados por los profesionales en EC1-PROF (ver Sección C.1), no se han incluido en esta sección.

Las imágenes escaneadas de los cuestionarios llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

D.3.1 Cuestionarios

Las figuras que se muestran a continuación (Figura D-19, Figura D-20, Figura D-21 y Figura D-22) presentan los cuestionarios que tuvieron que llenar los participantes al estudio.

Cuestionario Ventas

Nombre y apellidos: _____

De acuerdo a su experiencia y considerando que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

1. ¿Construiría primero "cerrar caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
2. ¿Construiría primero "mantener caja" antes de "abrir caja"?
a) Sí b) No c) Me es indiferente
3. ¿Construiría primero "mantener notas de pedido" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
4. ¿Construiría primero "visualizar reporte de consolidado de ventas diarias" antes de "registrar venta"?
a) Sí b) No c) Me es indiferente
5. ¿Construiría primero "registrar venta" antes de "cerrar caja"?
a) Sí b) No c) Me es indiferente
6. ¿Construiría primero "mantener mesas" antes de "mantener notas de pedido"?
a) Sí b) No c) Me es indiferente
7. ¿Construiría primero "mantener caja" antes de "visualizar reporte de consolidado de ventas diarias"?
a) Sí b) No c) Me es indiferente

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura D-19. Cuestionario 1 para EC2-POST

Cuestionario Colegio

Nombre y apellidos: _____

De acuerdo al UCPD y que no tiene ninguna restricción por parte del usuario para construir el sistema, responda a las siguientes preguntas:

1. ¿Construiría primero "registrar matrícula" antes de "registrar grados y secciones"?
a) Sí b) No c) Me es indiferente
2. ¿Construiría primero "registrar pago de matrícula" antes de "registrar matrícula"?
a) Sí b) No c) Me es indiferente
3. ¿Construiría primero "realizar mantenimiento de alumnos" antes de "registrar matrícula"?
a) Sí b) No c) Me es indiferente
4. ¿Construiría primero "emitir reporte de alumnos morosos" antes de "realizar mantenimiento de alumnos"?
a) Sí b) No c) Me es indiferente
5. ¿Construiría primero "emitir constancia de matrícula" antes de "registrar pago de matrícula"?
a) Sí b) No c) Me es indiferente
6. ¿Construiría primero "generar boletas de pago" antes de "registrar pago de matrícula"?
a) Sí b) No c) Me es indiferente
7. ¿Construiría primero "emitir reporte de alumnos morosos" antes de "registrar pago de matrícula"?
a) Sí b) No c) Me es indiferente

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-4051

Figura D-20. Cuestionario 2 para EC2-POST

Cuestionario 3

Nombre y apellidos: _____

CONSTRUCCIÓN DE CASOS DE USO

1. Indique la secuencia que seguiría para construir (diseño, programación y pruebas) el mantenimiento de maestros, reportes y transacciones

a) Maestros-transacciones-reportes b) Maestros-reportes-transacciones c) Reportes-maestros-transacciones
d) Reportes-transacciones-maestros e) Transacciones-reportes-maestros f) Transacciones-maestros-reportes

DIAGRAMA DE PRECEDENCIAS

	-	0	1	2	3	4	+
2. Facilidad para construir o realizar el diagrama de precedencias	0	1	2	3	4		
3. Utilidad del diagrama de precedencias para determinar la secuencia de construcción de software	0	1	2	3	4		
4. Para los proyectos que haya realizado, ¿le hubiese parecido muy complicado el construir el software sin seguir la secuencia sugerida por el diagrama de precedencias? (0=nada complicado, 4=muy complicado)	0	1	2	3	4		
5. ¿Usaría el diagrama de precedencias para siguientes proyectos de desarrollo de software?	0	1	2	3	4		

COMENTARIOS Y SUGERENCIAS DEL DIAGRAMA DE PRECEDENCIAS

José Antonio Pow-Sang, Pontificia Universidad Católica del Perú, Proyecto DAM-ROS

Figura D-21. Cuestionario 3 para EC2-POST

D.3.2 Tablas con información obtenida de los cuestionarios

La Tabla D-6 y la Tabla D-7 muestran los resultados obtenidos en los tres primeros cuestionarios del estudio. Un valor de “1” indica que acertó en la respuesta y “0” es respuesta errónea.

Tabla D-6. Resultados obtenidos en el cuestionario 1 de EC2-POST

Participante	Pregunta							Cantidad de Respuestas correctas
	1	2	3	4	5	6	7	
1	1	0	0	1	1	0	0	3
2	1	0	0	1	1	1	0	4
3	1	0	0	1	0	0	1	3
4	0	1	0	1	1	0	1	4
5	1	0	0	1	1	0	0	3
6	1	0	1	1	1	1	1	6
7	1	0	1	1	0	0	1	4
8	1	1	1	1	1	0	1	6
9	1	0	1	1	1	1	0	5
10	1	1	0	0	1	0	0	3
11	0	1	0	1	0	0	1	3
12	1	1	1	1	1	1	1	7
13	1	0	0	0	1	0	1	3
14	1	0	0	0	1	0	0	2

Tabla D-7. Resultados obtenidos en el cuestionario 2 de EC2-POST

Participante	Pregunta							Cantidad de Respuestas correctas
	1	2	3	4	5	6	7	
1	1	1	1	1	1	1	1	7
2	1	1	1	1	1	1	1	7
3	1	1	1	0	1	1	1	6
4	1	0	1	1	1	0	1	5
5	1	1	1	1	1	1	1	7
6	1	0	1	1	1	1	1	6
7	1	1	1	1	1	1	1	7
8	1	1	1	1	1	1	1	7
9	1	1	1	1	1	1	1	7
10	1	1	1	1	0	0	1	5
11	1	1	0	0	0	0	0	2
12	1	0	1	1	0	1	1	5
13	1	1	0	1	1	1	1	6
14	0	1	0	1	1	1	1	5

La Tabla D-9 muestra el resultado obtenido en el cuestionario 3 de EC2-POST. Cabe resaltar que a diferencia del experimento EC1-PROF, todos los participantes seleccionaron que la secuencia más fácil para construir un software es maestros-transacciones-reportes.

Tabla D-9. Resultados obtenidos en el cuestionario 3 de EC2-POST

Participante	Pregunta		
	2	3	5
1	3	3	3
2	4	4	4
3	3	3	3
4	4	3	3
5	3	4	4
6	2	2	2
7	4	4	3
8	3	4	4
9	3	4	3
10	3	3	3
11	3	3	3
12	2	2	3
13	3	3	4
14	2	3	3

Para el procesamiento estadístico, se sumó un “1” a todos los valores de la tabla anterior, a fin de poder comparar los resultados que fueron obtenidos con los alumnos de pregrado.

D.4 Evaluación con Estudiantes de Pregrado: EC3-PREG

En esta sección se presenta la información adicional del estudio realizado con alumnos de pregrado EC3-PREG. Como este estudio fue una réplica del experimento controlado realizado con estudiantes de posgrado, estudio EC2-POST, no se incluyen en esta sección los casos de estudio y los cuestionarios, porque son los mismos que los aplicados en EC2-POST. Las imágenes escaneadas de los cuestionarios llenados por todos los participantes se encuentran disponibles en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

La Tabla D-10 y la Tabla D-11 muestran los resultados obtenidos en los tres primeros cuestionarios del estudio. Un valor de “1” indica que acertó en la respuesta y “0” es respuesta errónea.

Tabla D-10. Resultados obtenidos en el cuestionario 1 de EC3-PREG

Participante	Pregunta							Cantidad de Respuestas correctas
	1	2	3	4	5	6	7	
1	1	1	0	1	1	0	0	4
2	1	0	1	0	0	1	0	3
3	1	1	1	1	1	1	1	7
4	0	1	1	1	0	1	1	5
5	1	1	1	1	1	1	1	7
6	0	1	1	1	0	1	0	4
7	1	0	1	0	1	0	1	4
8	1	0	0	1	1	0	0	3
9	1	1	1	0	1	0	0	4
10	1	0	0	1	0	0	0	2
11	0	1	1	1	0	0	1	4
12	1	1	1	1	1	1	1	7
13	1	0	1	1	0	1	1	5
14	0	1	1	1	1	1	1	6
15	1	1	0	1	1	0	1	5
16	1	1	1	1	1	0	0	5
17	0	1	1	1	1	0	0	4
18	0	1	0	1	1	0	1	4
19	1	1	1	1	1	0	1	6
20	1	1	1	1	1	0	0	5
21	1	1	1	1	1	0	0	5
22	1	0	0	1	0	0	0	2
23	1	0	0	0	1	0	0	2
24	1	0	1	1	1	1	1	6
25	1	1	1	1	0	1	0	5
26	1	1	0	1	1	1	0	5
27	1	1	1	1	0	1	1	6
28	1	0	0	1	1	0	0	3
29	1	1	1	1	0	1	0	5
30	1	1	1	1	0	1	1	6
31	0	1	1	1	0	1	1	5
32	1	0	1	1	0	1	1	5
33	1	1	1	1	1	1	1	7
34	1	1	0	1	1	1	0	5
35	0	0	1	1	1	1	1	5

Tabla D-11. Resultados obtenidos en el cuestionario 2 de EC3-PREG

Participante	Pregunta							Cantidad de Respuestas correctas
	1	2	3	4	5	6	7	
1	1	1	1	1	1	1	1	7
2	1	1	1	1	1	1	1	7
3	1	1	0	1	1	1	1	6
4	1	1	1	1	1	0	1	6
5	0	1	1	1	1	1	1	6
6	1	1	1	1	1	1	0	6
7	1	0	1	1	1	1	1	6
8	1	1	1	1	1	0	1	6
9	1	1	0	1	1	1	1	6
10	1	1	0	1	1	1	1	6
11	0	1	0	1	1	1	1	5
12	1	1	1	1	1	1	1	7
13	1	0	1	1	1	1	1	6
14	1	0	1	1	1	1	1	6
15	1	1	1	1	1	1	1	7
16	1	1	0	1	0	1	1	5
17	1	1	0	1	1	0	0	4
18	1	1	1	1	1	1	1	7
19	1	1	1	1	1	1	1	7
20	1	1	1	1	1	0	0	5
21	1	1	1	1	1	1	1	7
22	1	1	0	0	1	1	0	4
23	1	1	0	1	1	0	1	5
24	1	1	1	1	1	0	1	6
25	1	1	1	1	1	1	1	7
26	1	1	0	1	1	1	1	6
27	1	1	1	1	1	1	1	7
28	1	1	1	1	1	1	1	7
29	1	1	1	1	1	1	1	7
30	1	0	0	0	1	1	1	4
31	1	1	1	1	1	1	1	7
32	1	1	0	1	0	NR	0	3
33	1	1	1	1	1	1	1	7
34	1	1	1	1	1	1	1	7
35	1	1	1	1	1	0	1	6

*NR significa que no respondió a la pregunta

La Tabla D-12 muestra el resultado obtenido en el cuestionario 3 de EC2-POST. Cabe resaltar que a diferencia del experimento EC1-PROF, todos los participantes seleccionaron que la secuencia más fácil para construir un software es maestros-transacciones-reportes.

Tabla D-12. Resultados obtenidos en el cuestionario 3 de EC3-PREG

Participante	Pregunta		
	2	3	5
1	3	3	3
2	3	3	2
3	2	4	3
4	2	4	4
5	4	3	3
6	3	3	3
7	3	3	3
8	3	4	4
9	4	4	4
10	3	2	1
11	2	3	3
12	4	4	4
13	3	4	4
14	4	2	3
15	3	4	4
16	3	3	3
17	3	4	4
18	4	4	4
19	3	4	4
20	2	4	3
21	3	4	2
22	3	3	4
23	2	4	3
24	4	2	4
25	3	4	4
26	3	4	4
27	3	4	4
28	3	3	4
29	2	3	4
30	3	4	3
31	3	4	4
32	2	3	3
33	3	3	3
34	3	4	4
35	3	4	3

Para el procesamiento estadístico, se sumó un “1” a todos los valores de la tabla anterior, a fin de poder comparar los resultados que fueron obtenidos con los alumnos de pregrado.

Información Complementaria del Capítulo 7

Anexo

E

En este anexo se incluye información complementaria relacionada a la evaluación de Incremental-FP (ver capítulo 7). La Sección E.1 muestra más detalles de la estimación realizada para el proyecto A del semestre 2004-2 y la Sección E.2, la estimación del proyecto B del semestre 2004-2.

La información detallada del resto de proyectos se encuentra disponible en el siguiente enlace <http://inform.pucp.edu.pe/~jpowsang/tupuy/comp>

E.1 Información del Proyecto A del 2004-2

La Tabla E-1 muestra el cálculo detallado de la cantidad de PFSA correspondientes al equipo A del semestre 2004-2.

Tabla E-1. Cálculo de PFSA del equipo A para el semestre 2004-1

Incr.	Caso de Uso	Transacción PF	PFSA por Transacción	PFS A por ILF	PFSA por Incr.
1	Login	Login (Caso de Uso Login de Usuario)	3	1.75	172.66
	Mantenimiento de clientes	Registro de nuevo cliente	4	0.91	
	Mantenimiento de clientes	Buscar cliente	3	0.91	
	Mantenimiento de clientes	Modificar cliente/eliminar	3	0.91	
	Registrar evaluación de riesgo	Registro de evaluación de riesgo	3	9.57	
	Registrar evaluación de riesgo	Consulta de producto de credito	3	1.40	
	Mantenimiento de productos de crédito	Añadir tipo de tarjeta de crédito	3	1.40	
	Mantenimiento de productos de crédito	Modificar datos de tipo de tarjeta	3	1.40	
	Mantenimiento de productos de crédito	Buscar tipo de tarjeta	3	1.40	
	Mantenimiento de productos de crédito	Buscar parámetro crediticio (combobox)	3	1.75	
	Mantenimiento de parámetros	Añadir parámetro crediticio	3	1.75	
	Mantenimiento de parámetros	Modificar parámetro crediticio	3	1.75	
	Mantenimiento de parámetros	Buscar parámetro crediticio	3	1.75	
	Mantenimiento de producto	Registra producto	3	0.50	
	Mantenimiento de producto	Buscar unidad (combobox)	3	2.33	
	Mantenimiento de producto	Buscar familia (combobox)	3	1.40	
	Mantenimiento de producto	Buscar proveedor (combobox)	3	1.17	

Anexo E. Información complementaria del capítulo 7

Incr.	Caso de Uso	Transacción PF	PFSA por Transacción	PFS A por ILF	PFSA por Incr.
	Mantenimiento de producto	Buscar marca (combobox)	3	1.75	
	Mantenimiento de producto	Modificar producto	3	0.50	
	Mantenimiento de transportistas	Registrar	3	1.40	
	Mantenimiento de transportistas	Modificar	3	1.40	
	Mantenimiento de transportistas	Buscar	3	1.40	
	Mantenimiento de unidad	Registrar	3	2.33	
	Mantenimiento de unidad	Modificar	3	2.33	
	Mantenimiento de familia	Registrar	3	1.40	
	Mantenimiento de familia	Modificar	3	1.40	
	Mantenimiento de familia	Buscar	3	1.40	
	Mantenimiento de marcas de producto	Registrar	3	1.75	
	Mantenimiento de marcas de producto	Modificar	3	1.75	
	Mantenimiento de marcas de producto	Buscar	3	1.75	
	Mantenimiento de sucursales	Registrar	3	1.40	
	Mantenimiento de sucursales	Modificar	3	1.40	
	Mantenimiento de sucursales	Buscar	3	1.40	
	Asignar cajeros	Asignar cajeros	3	0.00	
	Asignar cajeros	Combobox cajeros	3	2.45	
	Asignar montos iniciales	Asignar montos iniciales	3	0.70	
	Aperturar caja	Apertura de caja 1	3	0.70	
2	Mantenimiento de tarjetas adicionales	Añadir tarjeta adicional	3	1.17	208.22
	Mantenimiento de tarjetas adicionales	Modificar cliente adicional	3	1.17	
	Mantenimiento de tarjetas adicionales	Eliminar cliente adicional	3	1.17	
	Registrar refinanciamiento de crédito	Consulta de cliente y deudas	6	3.01	
	Registrar refinanciamiento de crédito	Ingresar refinanciamiento	3	0.70	
	Registrar pago	Consulta cliente y deuda	6	2.78	
	Registrar pago	Realizar pago	3	2.10	
	Consultas y Reportes	Reporte de clientes	4	1.61	
	Consultas y Reportes	Reporte de deudas	4	1.61	
	Consultas y Reportes	Reporte de cobranzas	4	1.61	
	Registrar entrada de productos	Registro de ingreso de almacén	4	4.90	
	Registrar entrada de productos	Consulta de orden de compra para registro	4	2.25	
	Registrar salida de productos	Registra de salida de almacen	6	11.90	
	Registrar salida de productos	Buscar producto	3	0.50	
	Mantenimiento de proveedores	Registro proveedor	4	1.17	
	Mantenimiento de proveedores	Modificar/eliminar proveedor	4	1.17	
	Mantenimiento de proveedores	Buscar proveedor	3	1.17	
	Generar pedido O/C	Registrar O/C	3	1.75	
	Generar pedido O/C	Buscar producto por proveedor	3	0.50	
	Generar pedido O/C	Buscar proveedor	3	1.17	

Incr.	Caso de Uso	Transacción PF	PFSA por Transacción	PFS A por ILF	PFSA por Incr.
	Generar pedido de compra	Consulta de O/C	6	3.42	
	Generar pedido de compra	Cambiar estado	3	1.75	
	Registrar y evaluar pedido de transferencia	Pedido de transferencia a otra tienda	3	2.33	
	Registrar y evaluar pedido de transferencia	Buscar producto por sucursal	3	0.50	
	Registrar y evaluar pedido de transferencia	Consultar pedidos de transferencia	6	4.23	
	Registrar y evaluar pedido de transferencia	Evaluar pedido de transferencia	3	2.33	
	Transferir efectivo	Transferencia de efectivo	3	0.70	
	Transferir efectivo	Combobox caja	3	0.70	
	Registrar venta	Registrar venta	4	2.25	
	Registrar venta	Registrar pago	4	6.57	
	Registrar reserva	Registro de reserva	4	11.91	
	Cerrar caja	Cierre de caja 1	3	2.45	
	Cerrar caja	Cierre de caja 2	4	0.70	
3	Consultas y Servicios	Consulta de Cuotas vencidas	3	0.70	86.12
	Consultas y Servicios	Cronograma de pagos	3	0.70	
	Consultas y Servicios	Consulta de crédito disponible	4	2.31	
	Consultas y Servicios	Retiro de efectivo	3	2.10	
	Manejo de imprevistos	Ingreso de imprevistos	3	7.00	
	Manejo de imprevistos	Buscar cliente	3	0.91	
	Registrar devolución de producto	Registrar devolución de producto	3	8.75	
	Registrar devolución de producto	Consulta de venta	4	2.25	
	Generar reporte de productos de bajo stock	Productos con bajo stock	3	0.50	
	Arqueo de caja	Consulta de cierre de caja	5	0.70	
	Reporte de ventas	Reporte de cajeros	3	1.75	
	Reporte de ventas	Reporte de cajas	3	0.70	
	Reporte de ventas	Reporte de ventas	7	5.05	
	Reportes de ventas	Reporte de entrada y salida de dinero	5	0.70	

E.2 Información del Proyecto B del 2004-2

La Tabla E-2 muestra el cálculo detallado de la cantidad de PFSA correspondientes al equipo A del semestre 2004-2.

Tabla E-2. Cálculo de PFSA del equipo B para el semestre 2004-1

Incr	Caso de Uso	Transacción PF	PFSA por Transacción	PFSA por ILF	PFSA por Incremento
1	Actualizar Beneficios de Plan	Actualizar Beneficios de Plan	6	5.25	159.13
	Asistir en la clasificación de cliente	Otorgamiento de Crédito 1	4	0.79	
	Asistir en la clasificación de cliente	Otorgamiento de Crédito 2	3	0.79	
	Asistir en la clasificación de cliente	Otorgamiento de Crédito 3	4	2.54	
	Mantenimiento de Categoría Cliente	Mantenimiento de Categoría de Cliente	3	1.75	
	Mantenimiento de Plan	Mantenimiento de Plan	3	3.50	
	Registrar Parámetros de Gestón de Cobranza	Registrar Parámetros de Gestión de Cobranzas	3	7.00	
	Registrar Solicitud de Crédito	Registrar Solicitud de Crédito 1	4	0.79	
	Registrar Solicitud de Crédito	Registrar Solicitud de Crédito 2	4	0.79	
	Registrar Solicitud de Crédito	Registrar Solicitud de Crédito 3	3	0.79	
	Registrar Solicitud de Crédito	Registrar Solicitud de Crédito 4	3	0.79	
	Sugerir Categoría	Sugerir Categoría	3	1.75	
	Mantenimiento de Tienda	Añadir Tienda	3	0.70	
	Mantenimiento de Tienda	Eliminar Tienda	3	0.70	
	Mantenimiento de Tienda	Editar Tienda	3	0.70	
	Mantenimiento de productos	Añadir Productos	3	0.28	
	Mantenimiento de productos	Eliminar Productos	3	0.28	
	Mantenimiento de productos	Editar Productos	3	0.28	
	Mantenimiento de Proveedores	Añadir Proveedores	3	0.64	
	Mantenimiento de Proveedores	Editar Proveedores	3	0.64	
	Ajuste de Kardex	Ajustar Kardex	4	0.98	
	Reporte De Proveedores	Reporte Proveedores	4	0.92	
	Mantenimiento de Tienda	Consultar Tienda	3	0.70	
	Mantenimiento de Proveedor	Consultar Proveedor	3	0.64	
	Mantenimiento de Producto	Consultar producto	3	0.28	
	Mantenimiento de vendedor	Mantenimiento de Vendedor	4	3.50	
	Mantenimiento de vendedor	Búsqueda de Vendedor	3	5.25	
	Mantenimiento de Caja: Registrar Caj a	Registrar Caja	3	1.00	
	Registrar Venta	Registrar Venta: Principal	3	1.28	
	Registrar Venta	Registrar Venta: Vista Previa	4	1.28	
	Registrar Venta	Registrar Venta: Venta Registrada	6	5.57	

Incr	Caso de Uso	Transacción PF	PFSA por Transacción	PFSA por ILF	PFSA por Incremento
2	Mantenimiento de Tarjetas de Cliente	Mantenimiento de Tarjetas de Cliente	4	4.29	135.87
	Mantenimiento de Cliente	Mantenimiento de Cliente	4	0.79	
	Deshabilitar cuenta	Deshabilitar cuenta	3	0.88	
	Refinanciar Deuda	Refinanciar Deuda 1	4	1.66	
		Refinanciar Deuda 2	4	1.66	
	Buscar Cliente	Seleccionar Cliente	3	0.79	
	Buscar Cuenta	Seleccionar Cuenta	4	1.66	
	Mantenimiento de Catálogos	Añadir Catálogo	6	3.25	
		Editar Catálogo	6	3.25	
	Crear y Consultar Orden de Compra	Crear Orden de Compra	4	0.92	
	Mantenimiento de Catálogos	Consultar Catálogos	6	3.25	
	Crear y Consultar Orden de Compra	Consultar Orden de Compra	6	5.12	
	Crear Reporte de Productos	Reporte de Productos	3	0.28	
	Mantenimiento de Promoción	Añadir	6	2.03	
	Mantenimiento de Promoción	Modificar	6	2.03	
	Mantenimiento de Promoción	Eliminar	3	1.75	
	Anular Venta	Anular Venta	3	1.00	
	Mantenimiento de Caja (sin Registrar Venta)	Apertura de Caja	3	1.75	
	Mantenimiento de Caja (sin Registrar Venta)	Ajuste de Caja	3	1.75	
	Mantenimiento de Caja (sin Registrar Venta)	Cierre de Caja	3	1.75	
	Mantenimiento de Promoción	Búsqueda de Productos	3	0.28	
	Mantenimiento de Promoción	Búsqueda de Promoción	6	2.73	
3	Recategorizar Cliente	Recategorizar Cliente 1	4	0.79	91.00
	Recategorizar Cliente	Recategorizar Cliente 2	3	0.79	
	Recategorizar Cliente	Recategorizar Cliente 3	3	0.79	
	Consultar Cronograma de Pagos	Consultar Cronograma de Pagos	4	1.66	
	Consultar Clientes Morosos	Consultar Clientes Morosos	3	0.88	
	Consultar Estado de Cuenta	Consultar Estado de Cuenta	3	0.88	
	Crear y Consultar Nota de Ingreso	Crear Nota de Ingreso	6	10.25	
	Crear y consultar Guía Interna	Crear Guía Interna	6	3.31	
	Generar Reporte de Stock	Generar Reporte	3	0.98	
	Crear y Consultar Nota de Ingreso	Consultar Nota de Ingreso	6	4.42	
	Crear y consultar Guía Interna	Consultar Guía Interna	4	3.03	
	Registrar Pago de Cuotas	Registrar Pago de Cuotas	4	1.66	
	Generar Reportes	Reporte de Ventas	5	1.28	
	Generar Reportes	Reporte Histórico de Ventas	5	1.28	